Monitoring and Control of Renewable Energy Sources using PMUs

From Design through Implementation to Validation of Real-Time Synchrophasor-Based Tools

The Real-Smart Symposium

Nov. 29, 2013

Prof. Luigi Vanfretti

Associate Professor and Docent

KTH SmarTS Lab (luigiv@kth.se)

Special Advisor in Strategy and Public Affairs – R&D Division

Statnett SF (luigi.vanfretti@statnett.no)

SmarTS Lab

Smart Transmission Systems Laboratory

Statnett

Acknowledgement

- The work presented here is a result of the collaboration between KTH SmarTS Lab (Sweden), Statnett SF (Norway), and the IREC research center (Spain).
- This work has been financed by:
  - Statnett SF, the Norwegian transmission system operator, through its Smart Operation R&D program.
  - Nordic Energy Research through the STRONgrid project.
  - The European Institute of Technologies, EIT, through the Key Innovation Collocation Center InnoEnergy project “Smart Power”
  - The following people have contributed to this work:

Statnett

  - KTH SmarTS Lab: M. Shoaib Almas , Maxime Baudette, Dr. Iyad Al-Khatib, Viktor Appelgren
  - Statnett SF: Vemund H. Aarstrand, Stig Løvlund, Jan O. Gjerde
  - IREC: Ignasi Cairó, Jose Luís Dominguez, Gerard del Rosario, Alberto Ruiz
- Austin White of OG&E is gratefully acknowledged for providing PMU measurements from the Oklahoma power network which was used to develop our off-line tools.

Outline

- Motivation
  - Examples of challenges brought by renewable energy sources
  - Bringing renewable energy into the Nordic Grid – what can we expect?
  - From the X-Ray to the MRI to monitor and control renewable energy sources
- Methodology for Development of Real-Time PMU Apps in the Lab.
- Problem Statement – Identifying Wind Farm Induced Grid Interactions
- Application Design
- Development Tools for Implementation
- Prototype Implementation
- Testing – protocols and methods
- Validation – protocols and method
- Deployment

SmarTS Lab
Smart Transmission Systems Laboratory

Renewable energy sources bring more ***production*** capacity...

HOME PAGE | TODAY'S PAPER | VIDEO | MOST POPULAR | U.S. Edition

The New York Times — Business Day — Energy & Environment

WORLD | U.S. | N.Y. / REGION | BUSINESS | TECHNOLOGY | SCIENCE | HEALTH | SPORTS | OPINION

Search | Global | DealBook | Markets | Economy | Energy | Media

Intermittent Nature of Green Power Is Challenge for Utilities

Herb Swanson for The New York Times

Wind turbines atop the ridge line of Lowell Mountain in Vermont. The wind farm is owned by Green Mountain Power.

- But production from renewables is not enough to attain a sustainable energy system

- For a **secure, efficient, and flexible** use of renewables **safe power transmission** is necessary!

energy. Because energy produced by wind, for example, is intermittent, its generating capacity is harder to predict than conventional power's. And a lack of widely available, cost-effective ways to store electricity generated by wind only compounds the complex current marketplace.

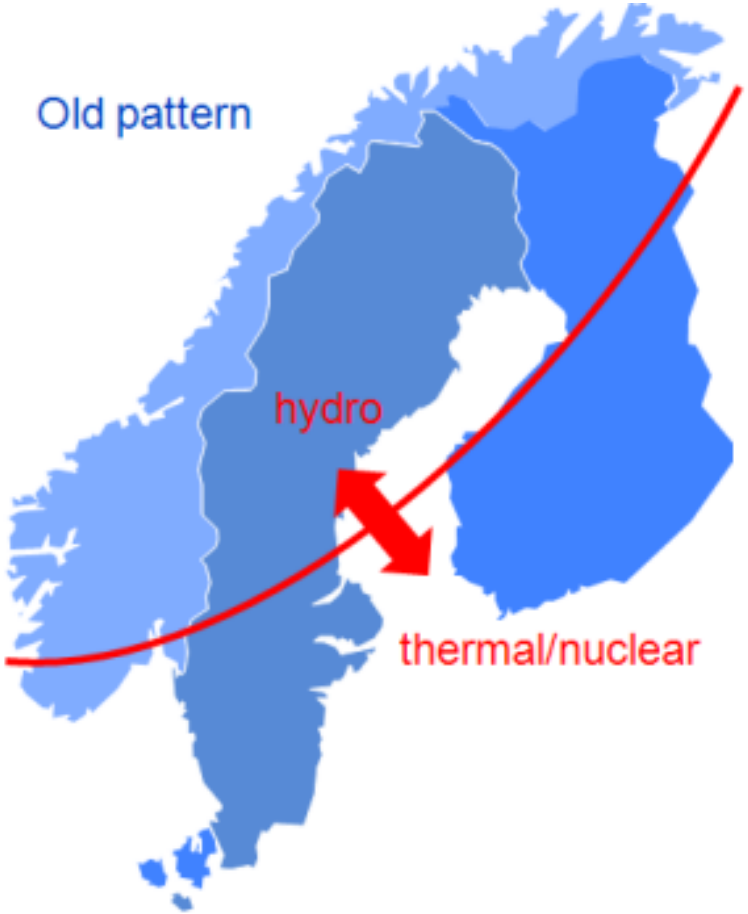
Moreover, although the wind industry has been growing for decades, it is still relatively new at operating large-scale wind farms, so it is often only once a farm is up and running that some of the problems emerge, developers say.

And those problems are likely to grow. Last year, wind power was the most prevalent source of new energy capacity — 43 percent of overall generation installed — while its price neared an all-time low, according to a recent report for the Department of Energy by Lawrence Berkeley National Laboratory.

A number of factors can trigger curtailments in wind output, including reducing the danger to bats or birds flying around the spinning blades. But more commonly, regional grid managers, who must match demand and supply instantaneously, call for a reduction in wind power when more energy is produced than the system can safely transport, they say.

Bringing More Renewable Capacity in the Nordic Grid

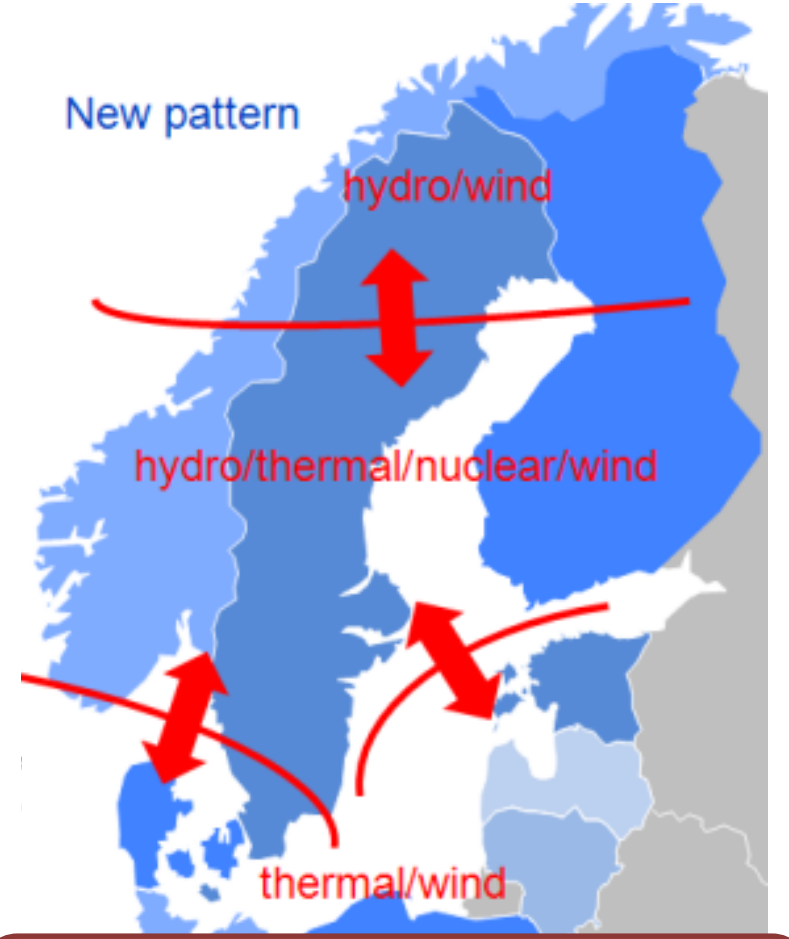
Will pose a challenges on how the grid is operated and controlled



Old pattern: winter/summer, wet/dry – mainly seasonal – operation based on historical experience!

Bringing More Renewable Capacity in the Nordic Grid

Will pose a challenges on how the grid is operated and controlled



New pattern: price differences, much wind / less wind – daily, hourly, per minute, per second changes!

- With changes in generation patterns – "system dynamics" will also change.
- This means *faster power transfer interactions* –which need to be **monitored, and controlled** for secure power transmission.
- ***Smart Grid solutions need to be developed to safely integrate renewable energy sources:***

  Real-time monitoring can provide real-time visibility and aid in assessing the **health** of the system due to fast dynamic phenomena – across traditional operational boundaries

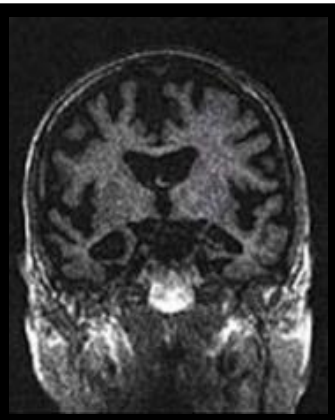
  Real-time control can help handling operation as operation conditions become more stringent

From the X-Ray to the MRI for the Power Grid

Identifying and controlling renewable energy sources in the grid

- We need advances in monitoring and control technology, similarly of the transition from the X-Ray to the MRI:

Grid monitoring technology of today

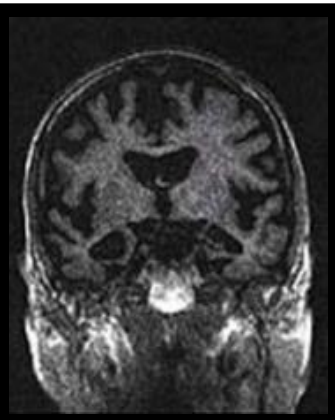


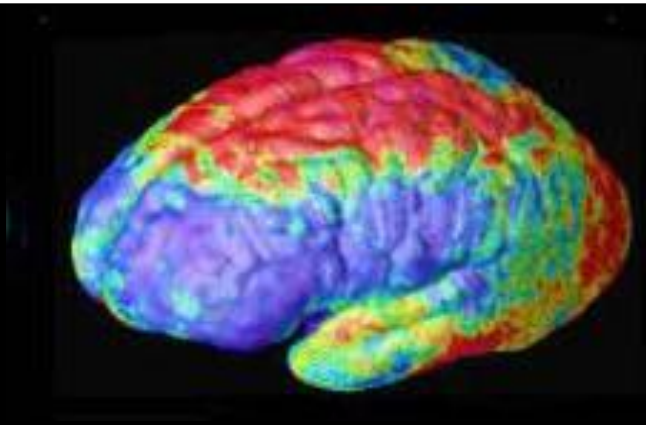
From the X-Ray to the MRI for the Power Grid

Identifying and controlling renewable energy sources in the grid

- We need advances in monitoring and control technology, similarly of the transition from the X-Ray to the MRI:

Grid monitoring technology of today



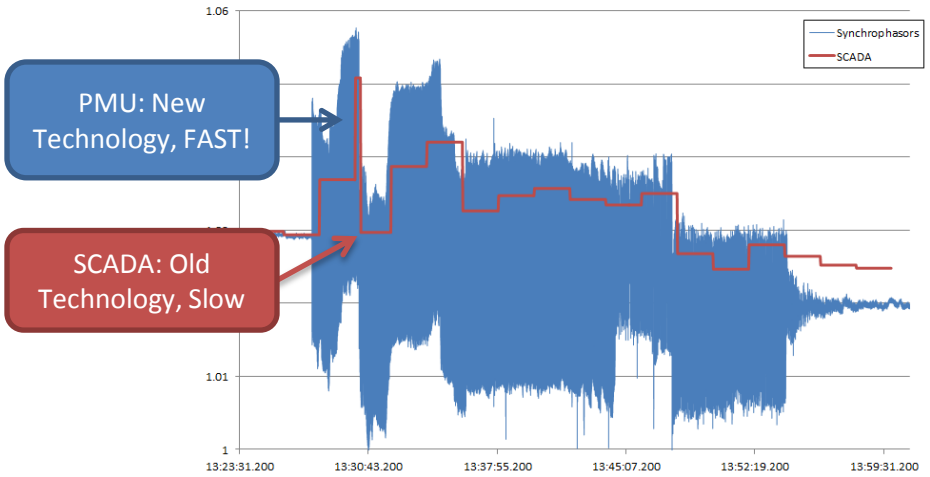


Smart Grid Monitoring Technology of tomorrow!

From the X-Ray to the MRI for the Power Grid

Identifying and controlling renewable energy sources in the grid

- We need advances in monitoring and control technology, similarly of the transition from the X-Ray to the MRI:



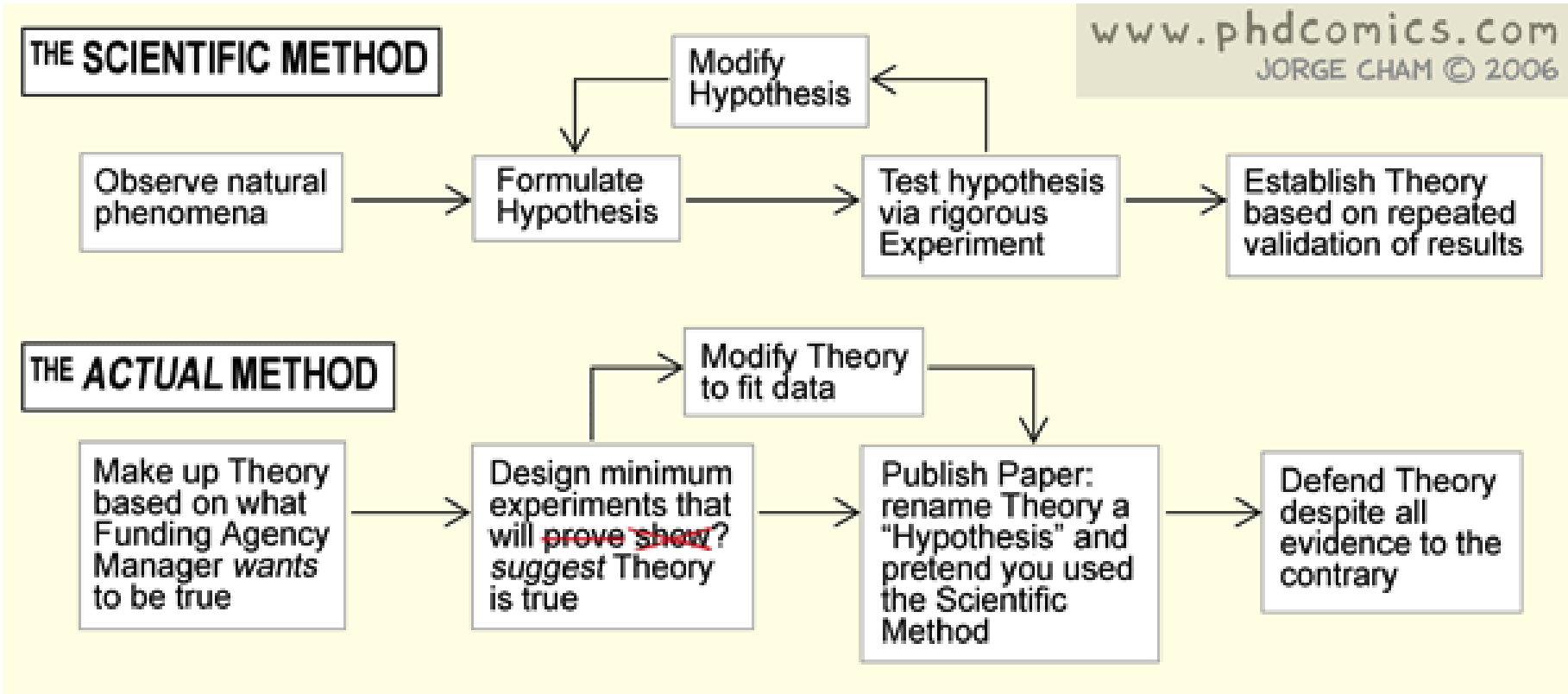
Wind-farm induced oscillations at @ OG&E

- Occurring during periods of high wind generation.
- 5% fluctuation at a frequency of 13-15 Hz.
- The oscillations were product of interactions between controllers in two different wind farms.
- Countermeasures:
  - Switching to electrically isolate the wind farms.
  - **Curtail the power output!**

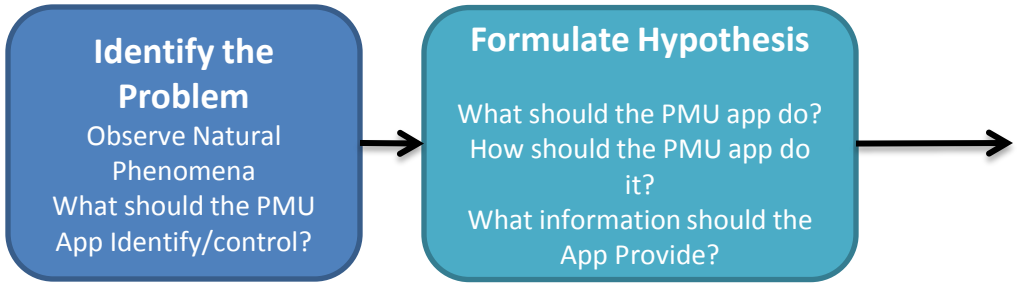
- Conventional Technology (SCADA) is not capable to cope with new challenges!: *too slow, can never capture the phenomena.*
- New software applications and power-electronics-based control technology can help to:
  - Detect unwanted behaviour from renewable energy sources (sub-synchronous wind-farm oscillations)
  - Control unwanted/unexpected behaviour and optimize the performance of the grid.
- **This in turn will allow to safely transport and utilize renewable energy sources!**

Methodology for Development of Real-Time Applications in the Lab.

Which method should we adopt to develop PMU Apps?


www.phdcomics.com
JORGE CHAM © 2006
THE SCIENTIFIC METHOD
Modify Hypothesis
Observe natural phenomena
Formulate Hypothesis
Test hypothesis via rigorous Experiment
Establish Theory based on repeated validation of results
THE ACTUAL METHOD
Modify Theory to fit data
Make up Theory based on what Funding Agency Manager wants to be true
Design minimum experiments that will prove shew? suggest Theory is true
Publish Paper: rename Theory a "Hypothesis" and pretend you used the Scientific Method
Defend Theory despite all evidence to the contrary

Methodology for Development of Real-Time Applications in the Lab.


Identify the Problem
Observe Natural Phenomena
What should the PMU App Identify/control?
Formulate Hypothesis
What should the PMU app do?
How should the PMU app do it?
What information should the App Provide?

Concept – Inception and Design of the PMU App

Formulate a hypothesis on how to solve the problem – what should the PMU app do? What are the requirements?

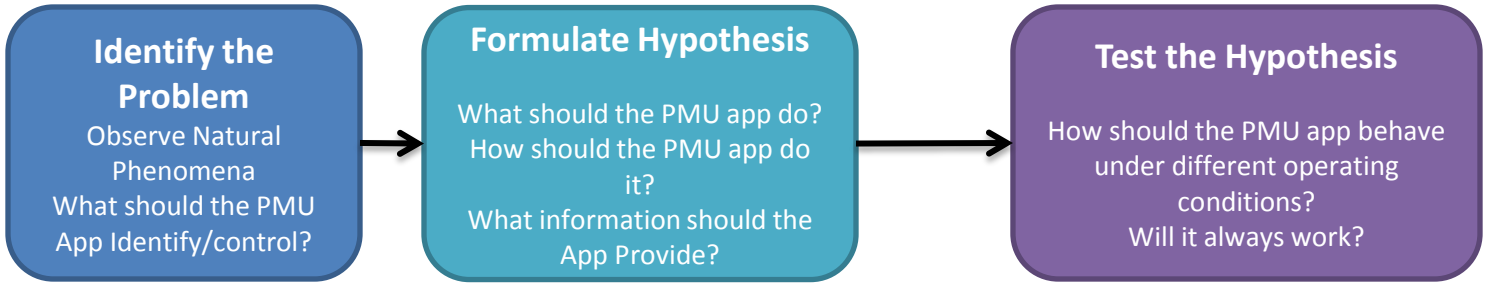
Methods

Design and implementation of algorithms (for monitoring, control, protection) with real-time execution constraints

Tools

SW architecture and prototype implementation of the proposed method to provide transparently the required information / action

Methodology for Development of Real-Time Applications in the Lab.

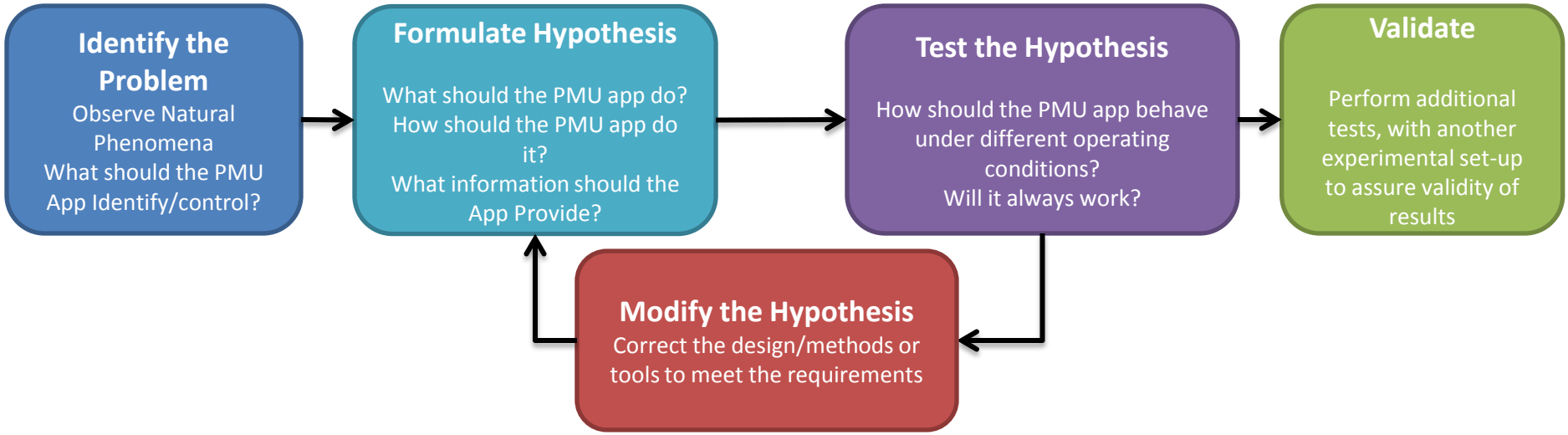

Identify the Problem
Observe Natural Phenomena
What should the PMU App Identify/control?
Formulate Hypothesis
What should the PMU app do?
How should the PMU app do it?
What information should the App Provide?
Test the Hypothesis
How should the PMU app behave under different operating conditions?
Will it always work?

Set-up Experimental Environment
Hardware and software needed to test the solution
Set-up the chain of real-time data acquisition

Design the Experiments Protocols
Develop an experiment protocol – i.e. what behaviour should be detected/control by the designed solution?

Models / HW Config.
Develop real-time simulation models that can reproduce the experiment protocol.
Set up hardware to reproduce the experiment protocol.

Methodology for Development of Real-Time Applications in the Lab.

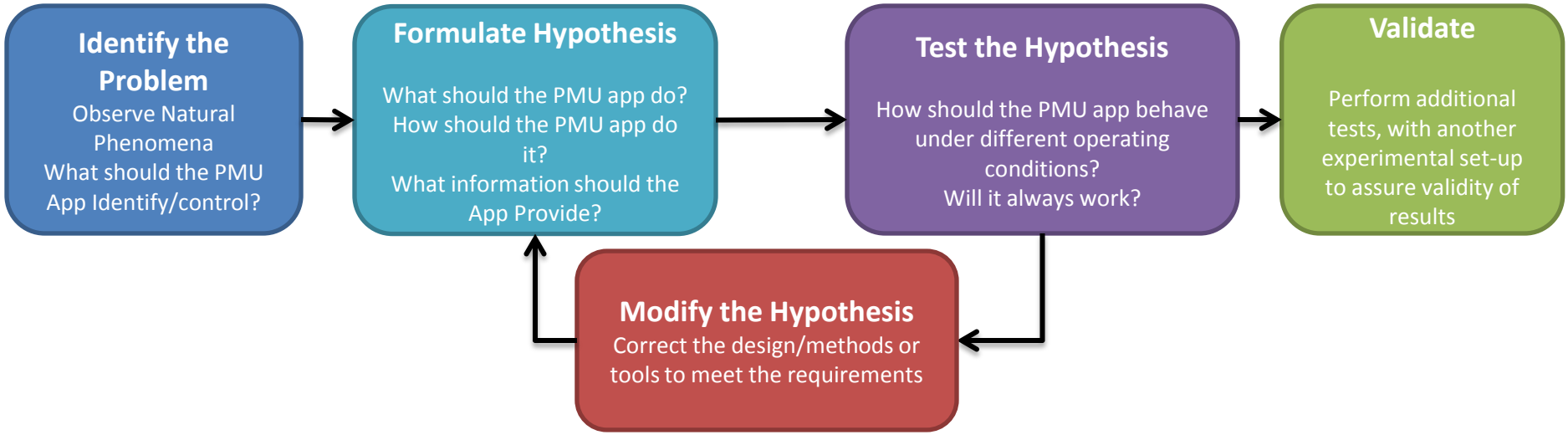

Identify the Problem
Observe Natural Phenomena
What should the PMU App Identify/control?
Formulate Hypothesis
What should the PMU app do?
How should the PMU app do it?
What information should the App Provide?
Test the Hypothesis
How should the PMU app behave under different operating conditions?
Will it always work?
Validate
Perform additional tests, with another experimental set-up to assure validity of results
Modify the Hypothesis
Correct the design/methods or tools to meet the requirements

Set-up Experimental Validation Environment
Configure the secondary experimental set-up for validation (different from the first) including the the chain of real-time data acquisition

Design the Validation Experiments Protocol
Develop new experiment protocols considering the properties/constraints of the validation environment

Gather Results and Determine Differences in Experiments
The PMU app should have performed well under the validation experiments. If not, find the source and go back to the hypothesis!

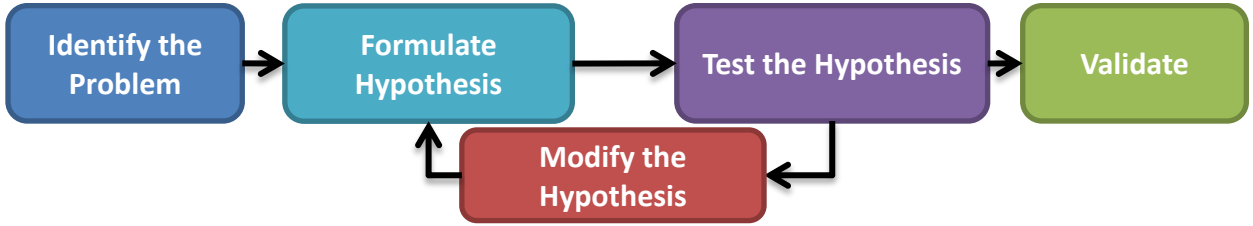
Methodology for Development of Real-Time Applications in the Lab.


Identify the Problem
Observe Natural Phenomena
What should the PMU App Identify/control?
Formulate Hypothesis
What should the PMU app do?
How should the PMU app do it?
What information should the App Provide?
Test the Hypothesis
How should the PMU app behave under different operating conditions?
Will it always work?
Validate
Perform additional tests, with another experimental set-up to assure validity of results
Modify the Hypothesis
Correct the design/methods or tools to meet the requirements

We will try to cover the different stages of the process and present the use of different tools for testing and validation in this presentation.

Shopping List

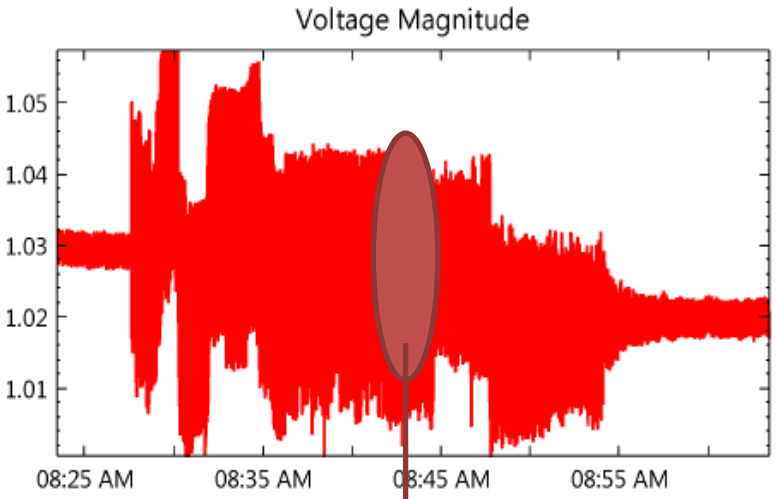
What do you need to develop real-time PMU Apps?



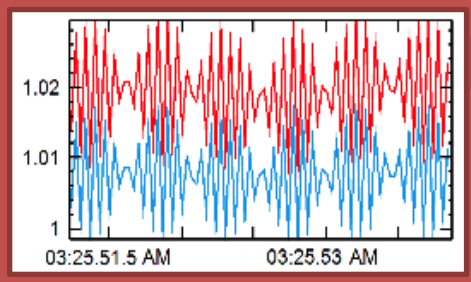
- Cost-free:
  - A *real problem* – what should the PMU App identify/control?
    - Don't make up a new problem that exists only in your head… we have plenty of things in reality
  - Good ideas on how to solve the problem:
    - You know what the PMU App should do and how it should do it.
- To develop the real-time PMU app and implement it:
  - You need software for real-time data mediation and handling.
  - You need a software development environment that lets you easily change things so that you can modify the hypothesis.
- To test the PMU App:
  - You need a real-time data source: PMUs either HW or virtualized.
  - You need a lab.: the real system (very risky), completely hardware-based (costly), real-time hardware-in-the-loop (kind of costly), other…
  - In RT HIL Simulation: you need a model that allows you to perform the tests that you need to show that the app works!
- To Validate the PMU App:
  - You need a second set up for validation – get some friend somewhere else to help you if you are not rich!

Problem Statement

Identify the Problem - Observation: Sub-Synchronous Wind-Farm Oscillations



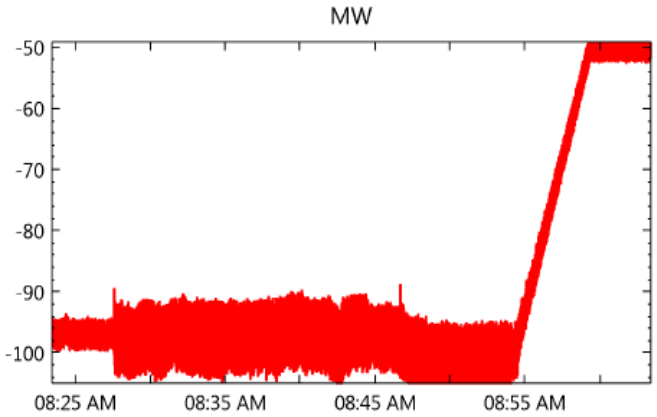
(a) Voltage Magnitude



PMU Data

~2 sec

- *Interaction is reflected in frequency components from 5 to 14 Hz.*
- ***Currently - The only means of mitigation is power output curtailment!***

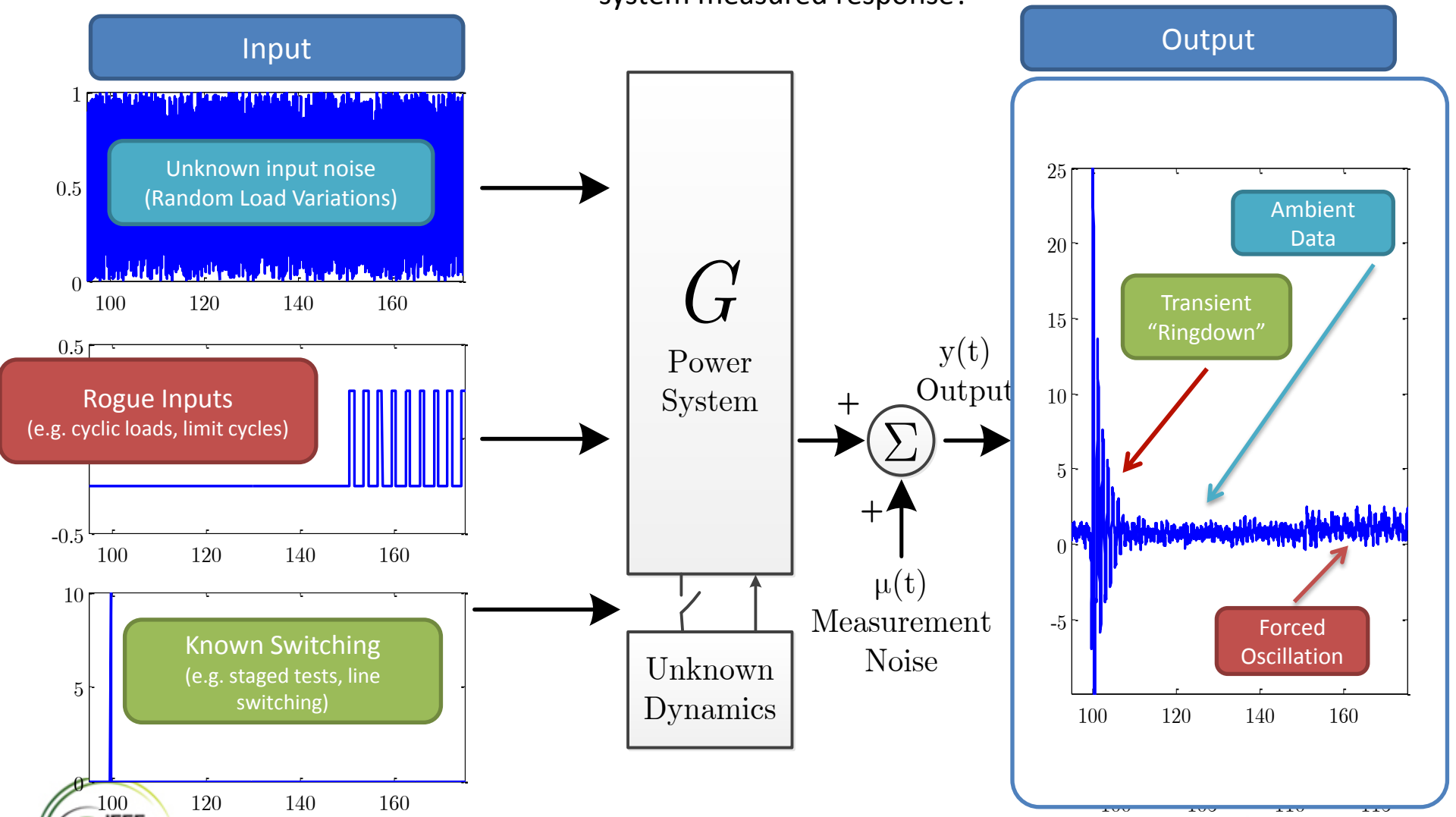


(b) Power Output

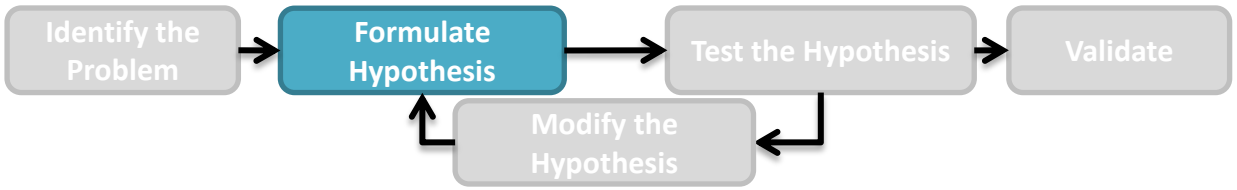
- **A new PMU App can be developed to detect sub-synchronous wind-farm oscillations at higher frequency than the traditional low-frequency oscillations.**
- **Means for detection is the first step towards control!**

Problem Statement

Identify the Problem - Observations: What information can be provided by a power system measured response?

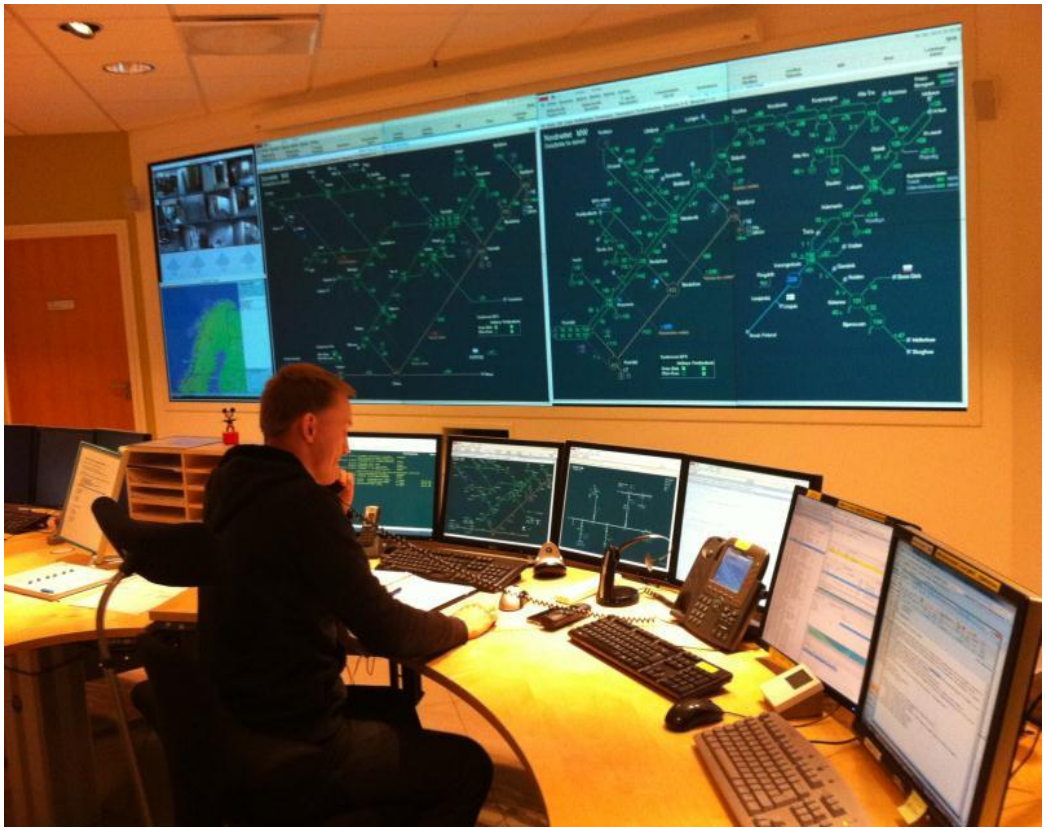


Formulate the Hypothesis

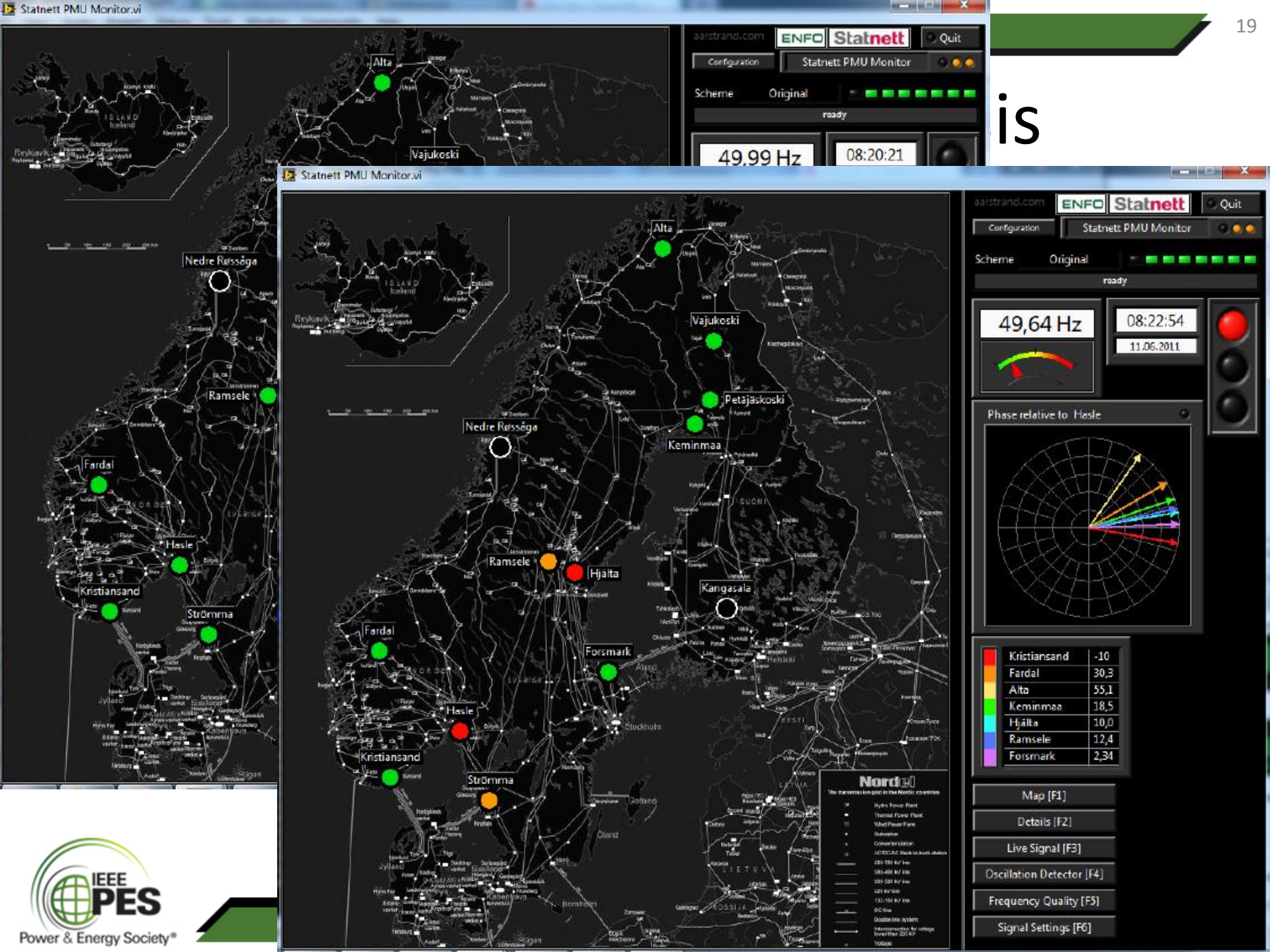


What should the PMU app do?
How should the PMU app do it?
What information should the App Provide?

Picture of the Regional Control Center at Alta. Statnett, Norway.

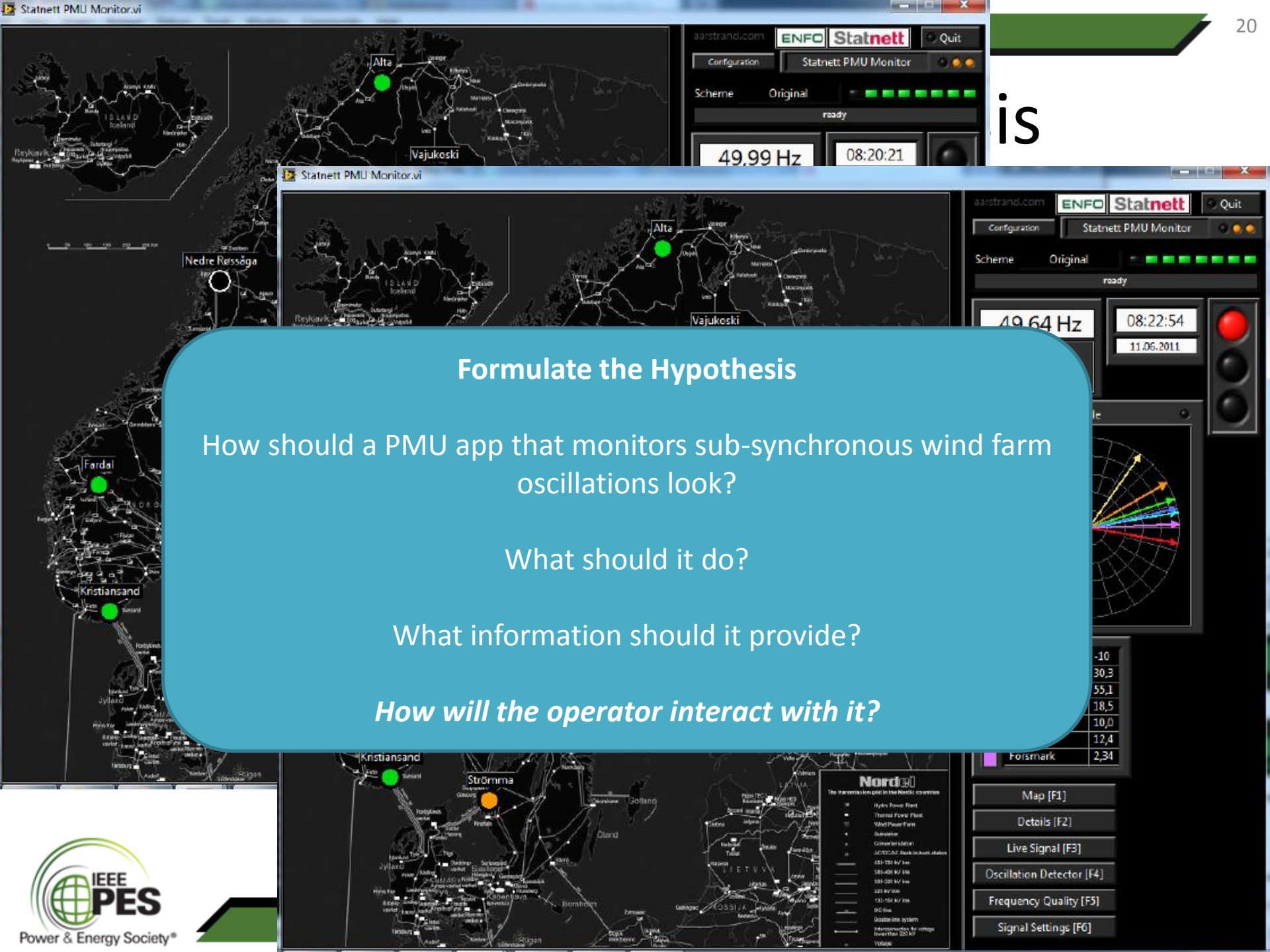


is



IEEE PES
Power & Energy Society®

is



Formulate the Hypothesis

How should a PMU app that monitors sub-synchronous wind farm oscillations look?

What should it do?

What information should it provide?

How will the operator interact with it?

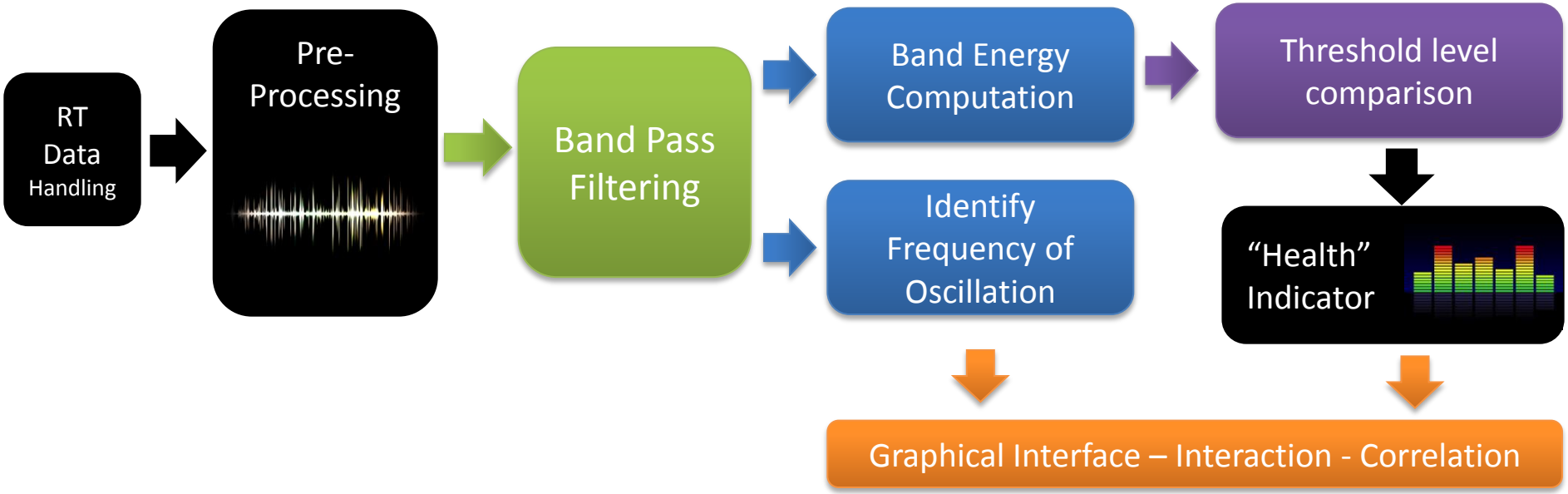
IEEE PES
Power & Energy Society®

Grid Equalizer

Concept

Inception and Design of the PMU App

- What should the App do?
  - The app should be able to detect oscillations at different frequency bands and provide a measure of the activity (health) of the oscillations at each band.
  - The app should help correlating the activity in each frequency band to a specific frequency of oscillation (or group of frequencies)
- What are the requirements?
  - Operate on real-time data streams and provide **fast** updates on the health indicator



Methods

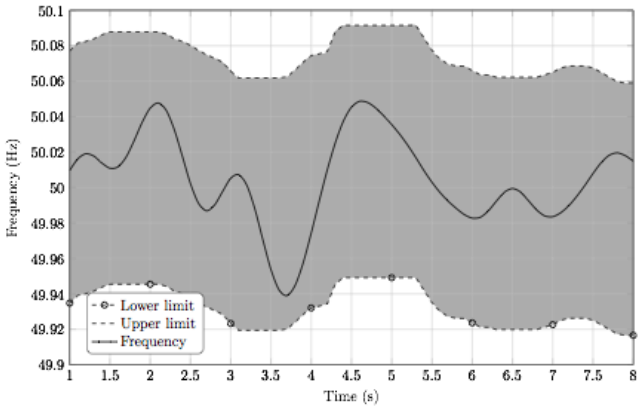
Methods

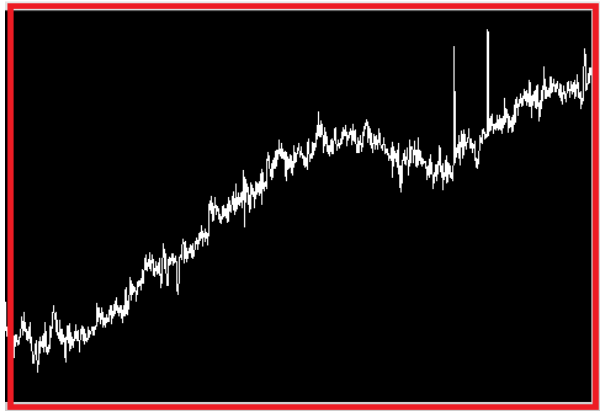
Design and implementation of algorithms (for monitoring, control, protection) with real-time execution constraints

- Design and Implementation of Algorithms:
  - Pre-processing: remove data flaws on the fly
  - Oscillation Detector:
    - A tool used to detect oscillatory activity – computes the energy of the oscillation within a given frequency band
    - Provides fast generalized alarm of oscillation activity in the freq. band
  - Spectral Estimator:
    - A tool used to estimate the spectrum of a signal
    - Uses digital signal processing methods – non-parametric and parametric methods
  - Fast RT Oscillation Detection and Monitoring Tool Components:
    - Real-time data stream display
      - Provides real-time overview of the selected measurement
    - Oscillation detectors at different and configurable ranges (low frequency inter-area modes, local modes, and fast modes (up to Nyquist freq.))
    - Detection and Alarming:
      - Energy of the oscillation at the given frequency
      - Alarm levels at configurable thresholds (ok, high activity, dangerous activity)

Pre-processing

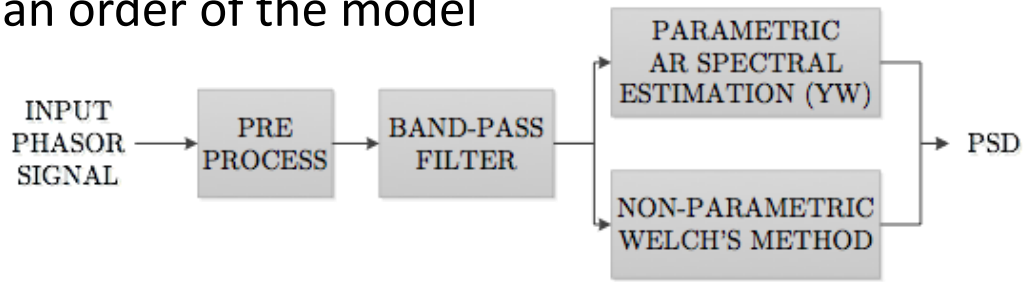
- Outlier removal for measurement errors
- Interpolation for signal outside of a confidence interval
  - Linear interpolation for avoiding divergences
- Implementation for the frequency:





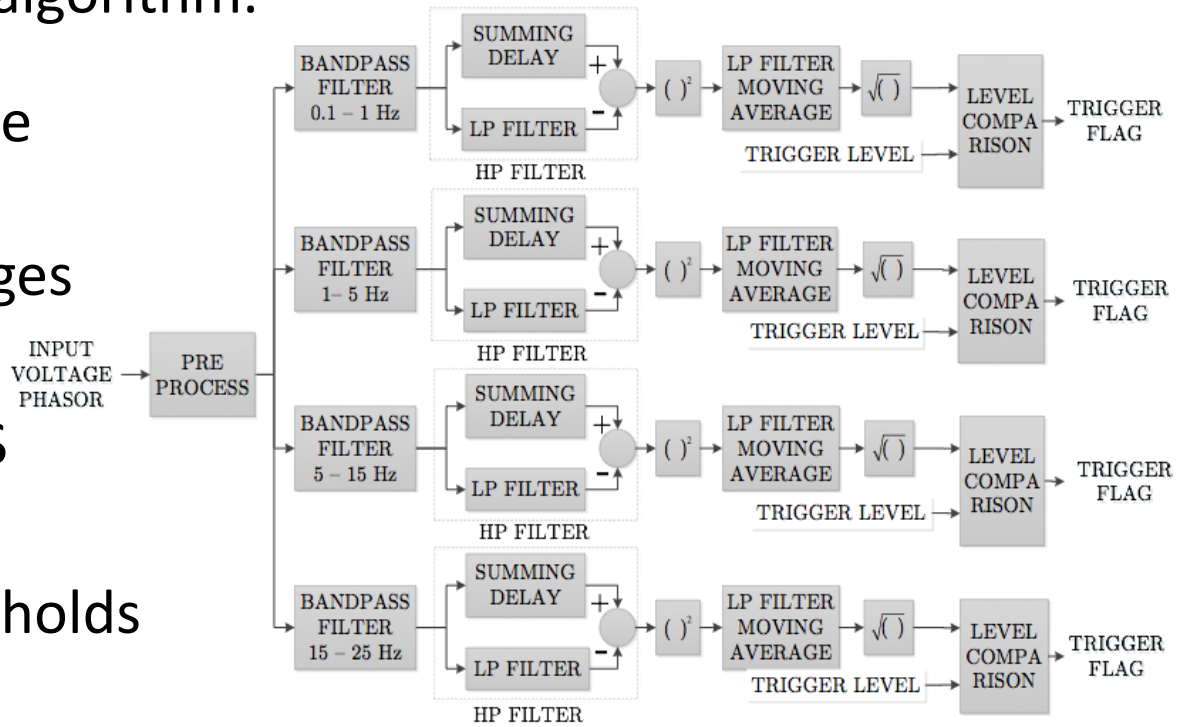
Spectral Estimator

- Four parallel algorithms automatically set to the same frequency ranges (for correlation with the oscillation detectors)
- Non-parametric method: Welch's method
  - Averaged modified Periodogram on overlapping windows
  - Useful if frequency content is unknown
- Parametric method: Auto-Regressive method
  - Curve fitting process, using a auto regressive model of the signal
  - Requires an order of the model



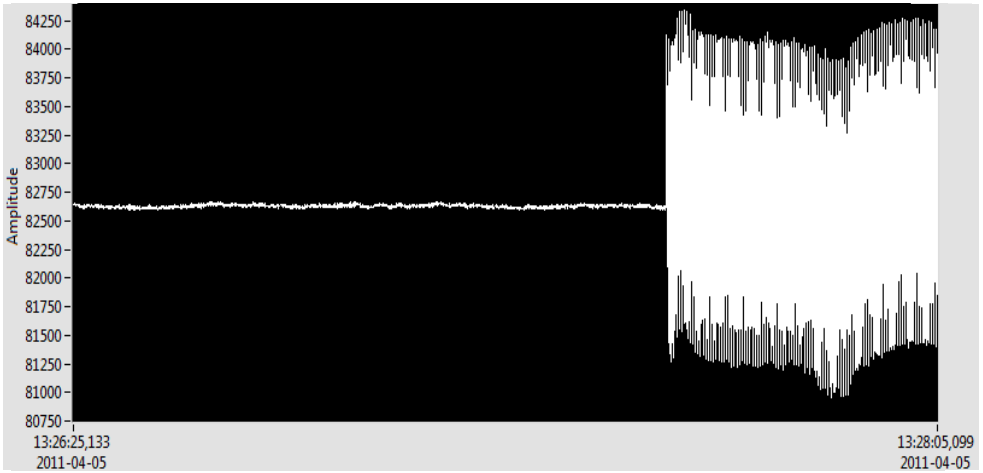
Oscillation Detector

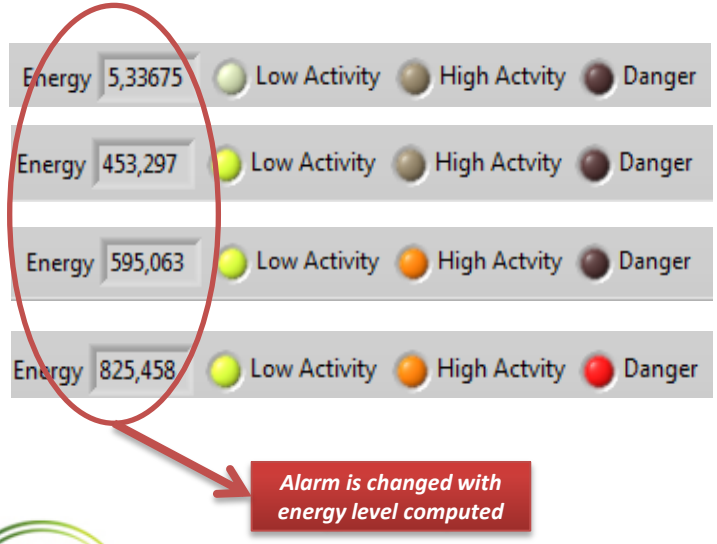
- Estimation of the activity of oscillatory components in a given frequency range.
- Envelope detection algorithm.
- Multiple configurable frequency ranges (Recommended ranges presented here)
- Modification to RMS calculations
- 3 Configurable thresholds

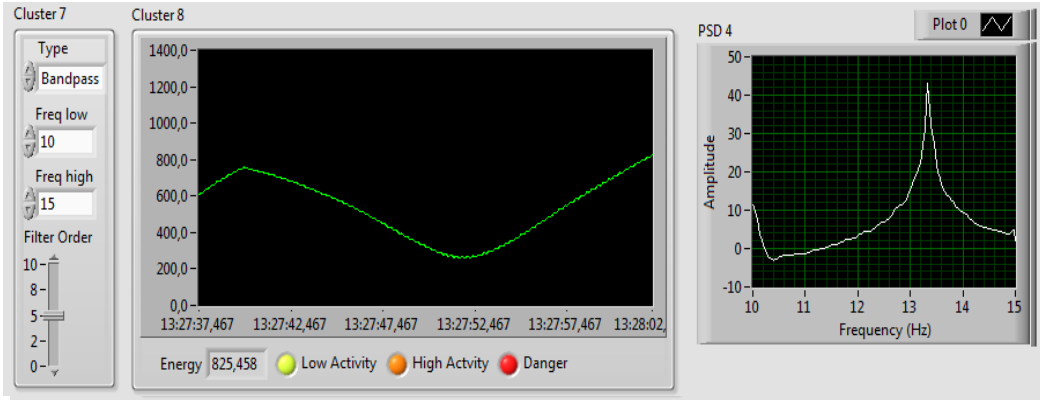


Generation of "Health" Signals

Threshold Level Comparison



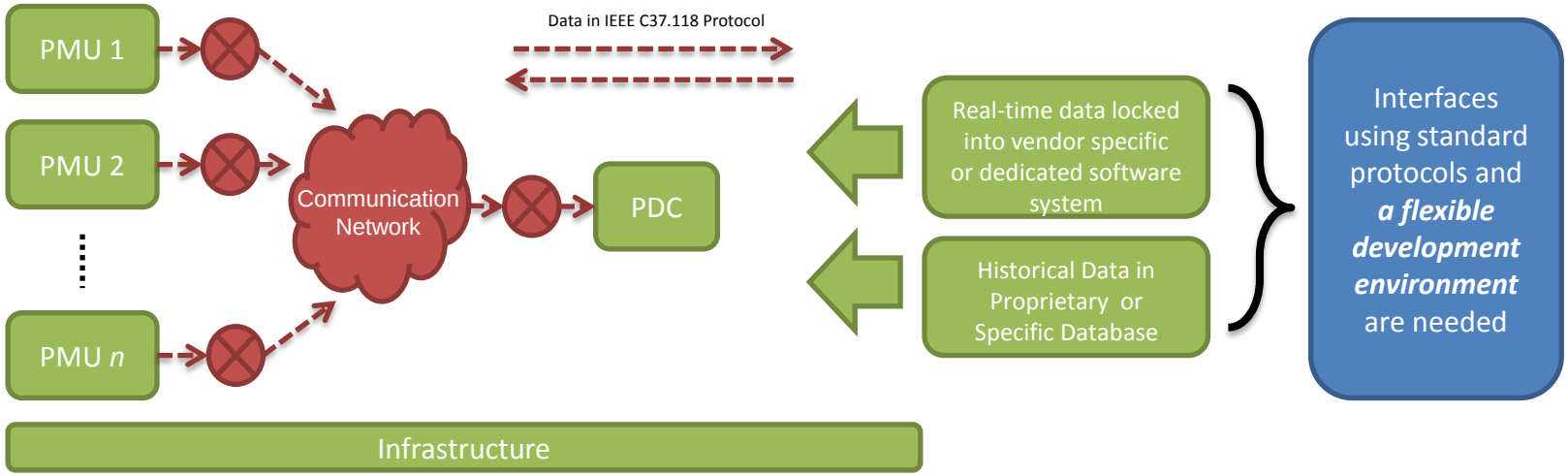

Energy 5,33675 Low Activity High Actvity Danger
Energy 453,297 Low Activity High Actvity Danger
Energy 595,063 Low Activity High Actvity Danger
Energy 825,458 Low Activity High Actvity Danger
Alarm is changed with energy level computed



Tools
SW architecture and prototype implementation of the proposed method to provide transparently the required information / action

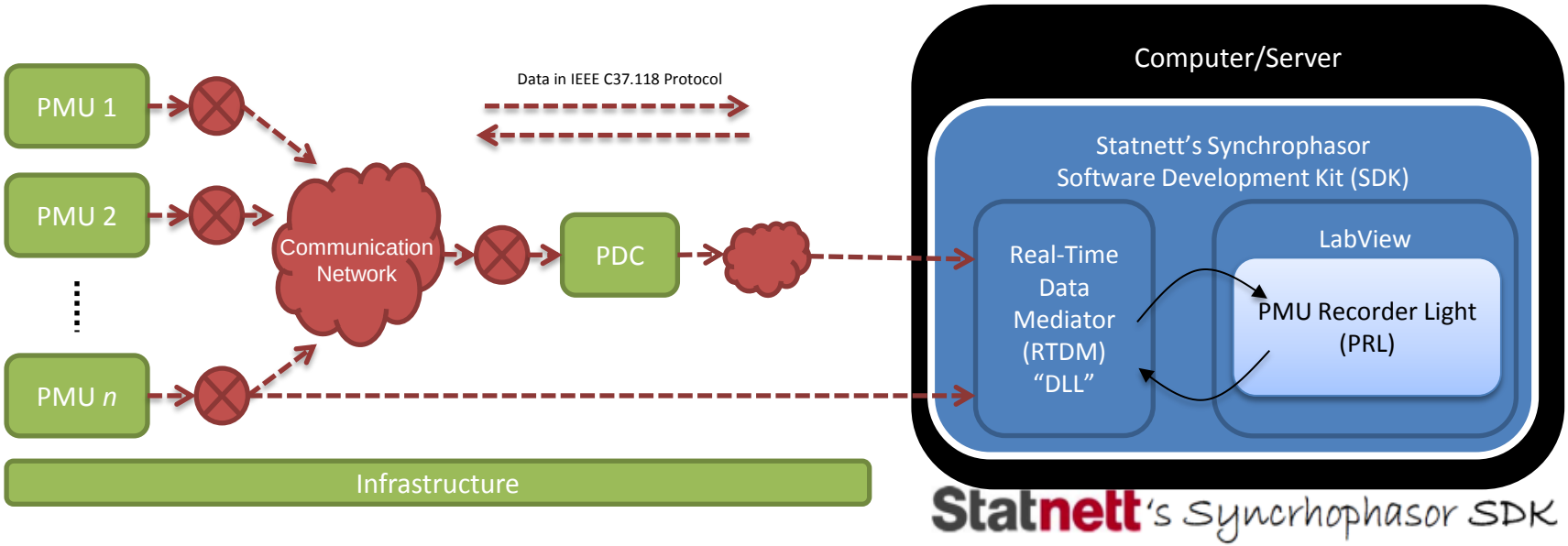
Tools

Building PMU Apps. “the last mile”: *the problem*


PMU 1
PMU 2
PMU n
Communication Network
PDC
Data in IEEE C37.118 Protocol
Real-time data locked into vendor specific or dedicated software system
Historical Data in Proprietary or Specific Database
Interfaces using standard protocols and ***a flexible development environment*** are needed
Infrastructure

Statnett's Synchrophasor

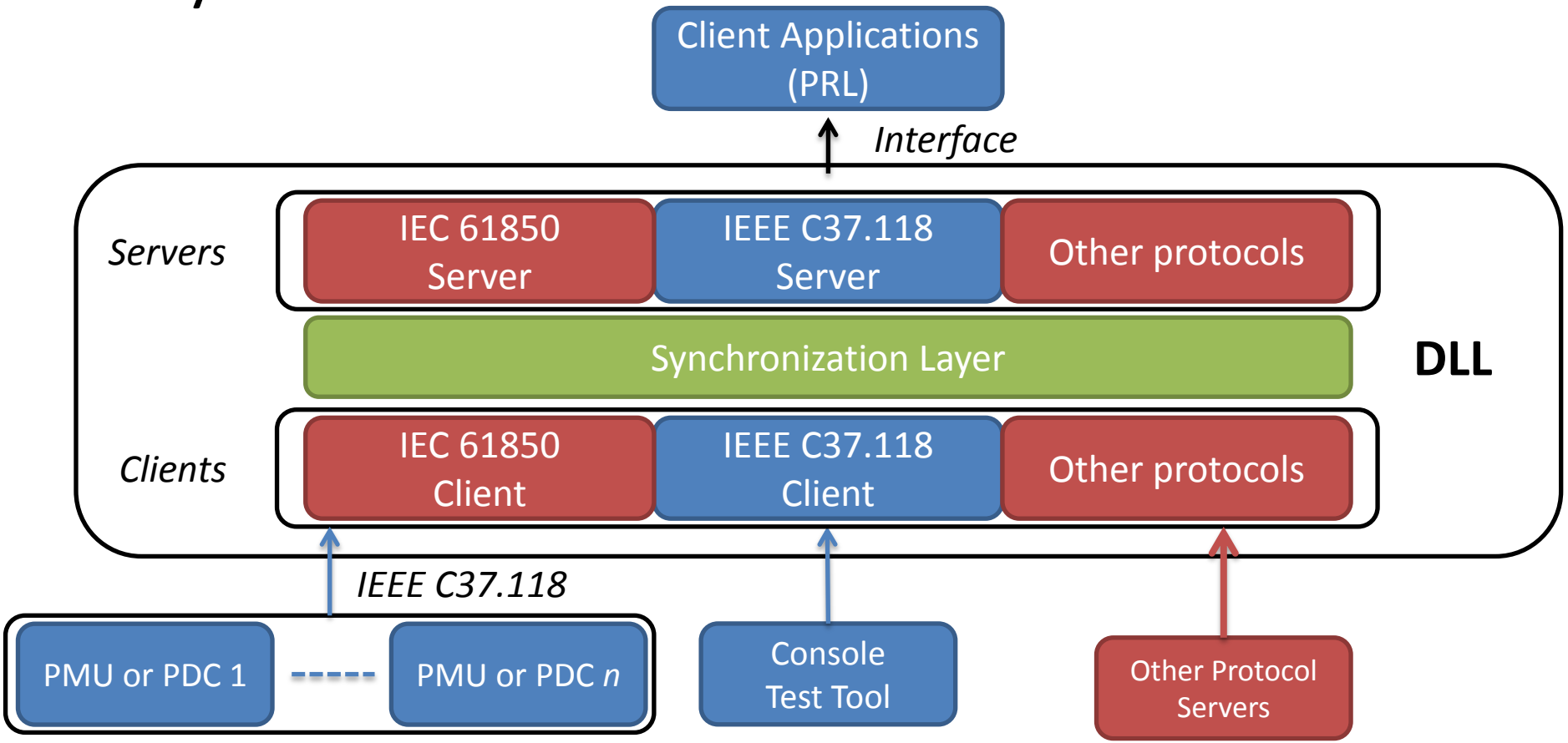
Software Development Kit (SDK)



- Infrastructure (external to the software development kit):
  - PMU: phasor measurement unit. Instrument providing GPS-time stamped measurements of voltage, current, and other quantities.
  - PDC: phasor data concentrator. A software running in a dedicated server
  - Communication Network: composed by routers/switches, fiber optic links (or other medium)
- Software Development Kit (SDK): a set of different computer software that allows a user to develop other derived software applications.
- Our SDK is composed by two main pieces:
  - Real-Time Data Mediator (DLL)
  - PMU Recorder Light (PRL)

Real-Time Data Mediator a.k.a. the "DLL"

- System Architecture

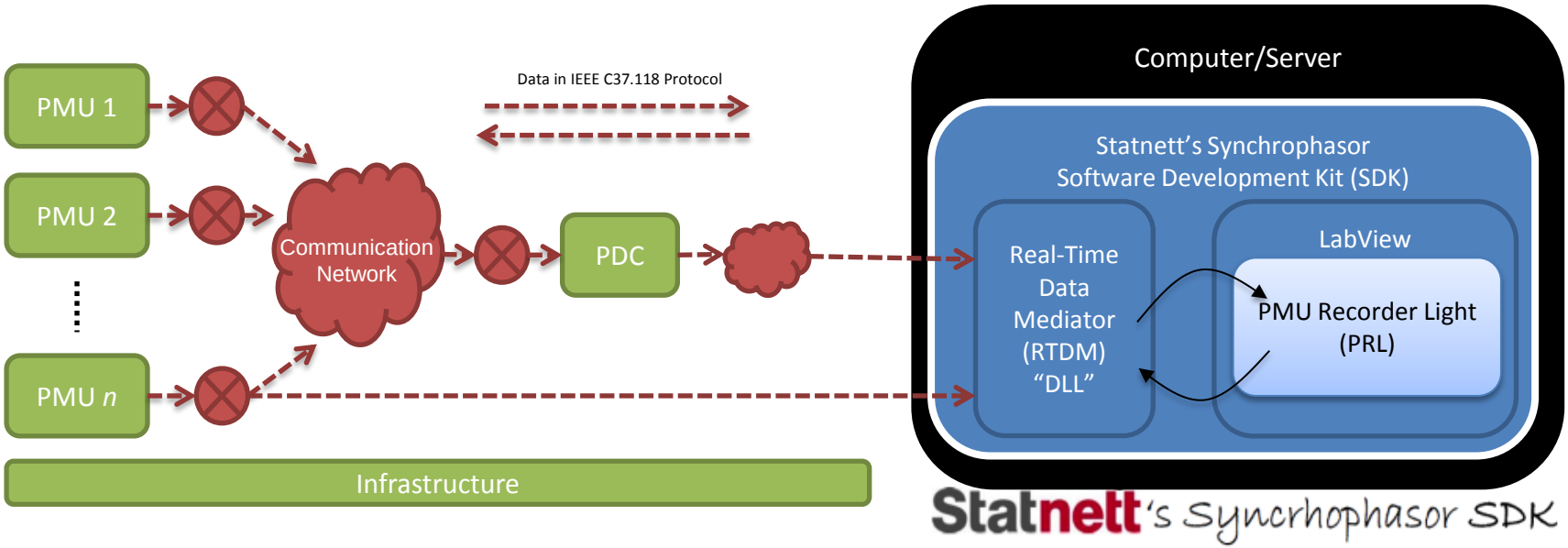


Only prepared, not fully implemented.

Implemented and Tested

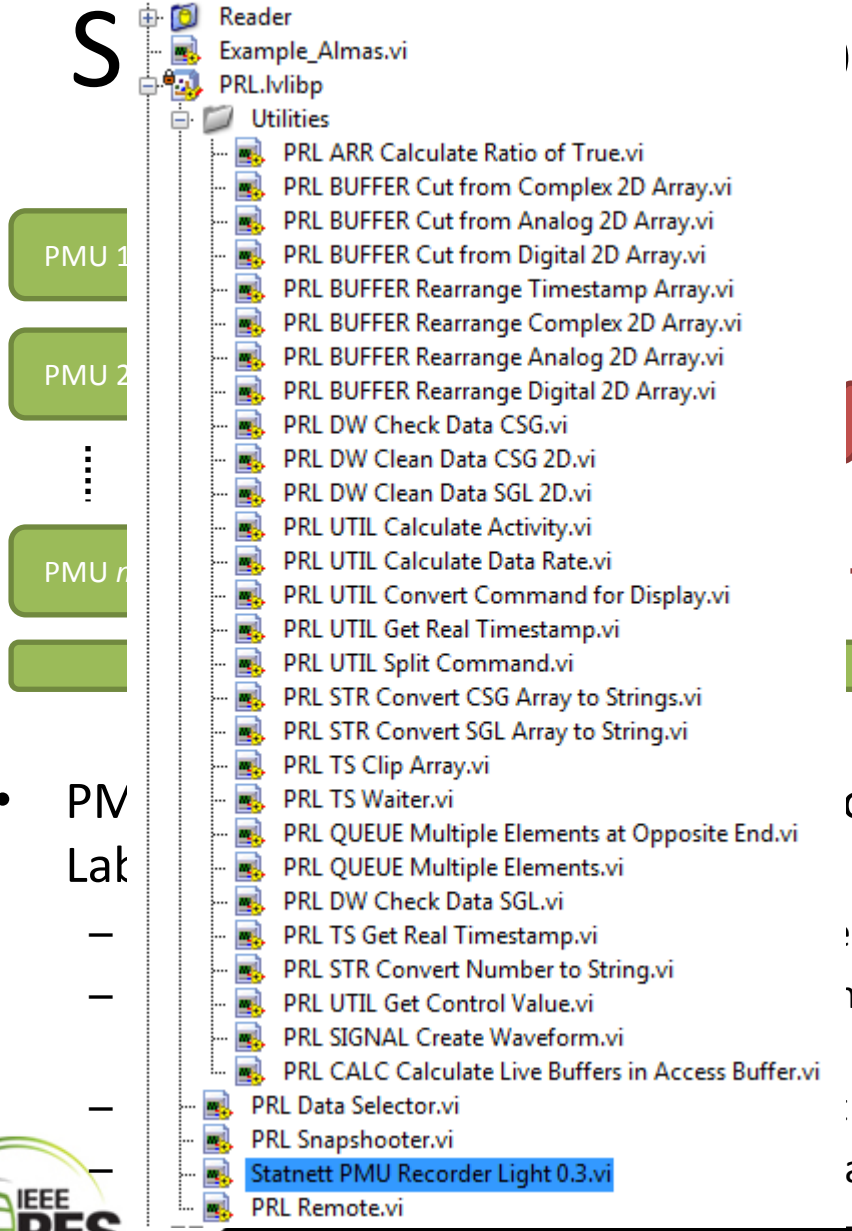
Not implemented

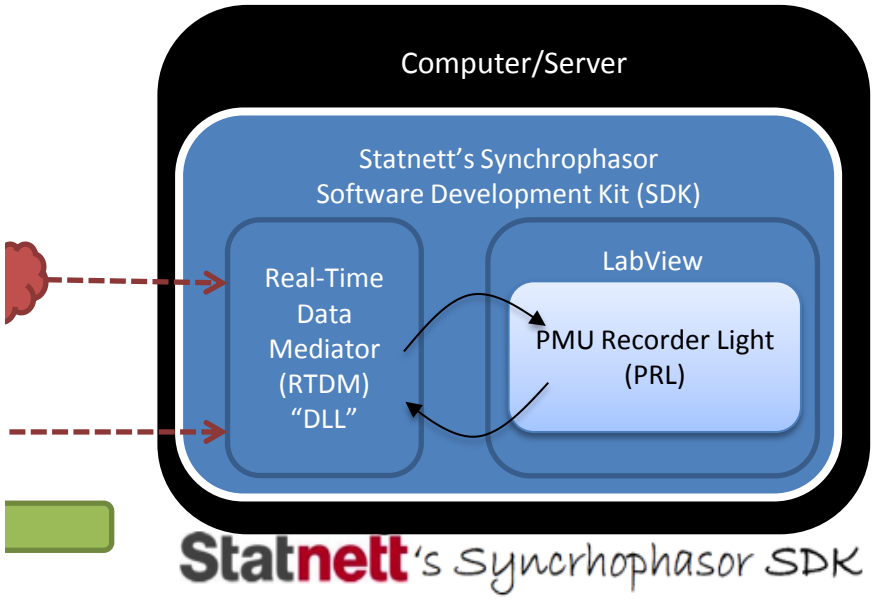
Software Development Kit (SDK)



- PMU Recorder Ligth (PRL): a set of computer programs written in the LabView language that facilitate the following
  - Interface with the RTDM for obtaining real-time PMU data
  - Providing graphical configuration of connections to PMUs and PDCs to be sent by the RTDM
  - Different function (graphical blocks) that a user can utilize to make a new program
  - Other functionalities listed in user manual and internal documents.

S pment Kit (SDK)

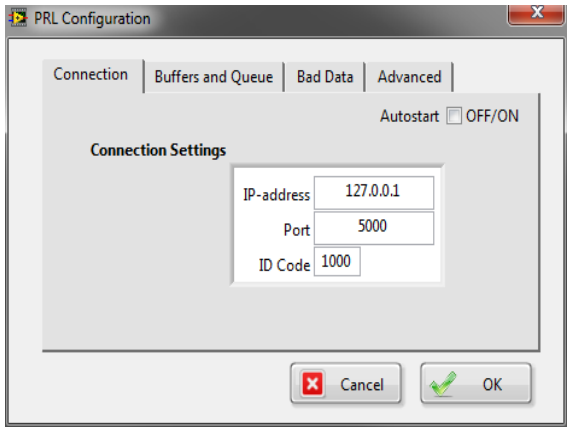

Reader
Example_Almas.vi
PRL.lvlibp
Utilities
PRL ARR Calculate Ratio of True.vi
PRL BUFFER Cut from Complex 2D Array.vi
PRL BUFFER Cut from Analog 2D Array.vi
PRL BUFFER Cut from Digital 2D Array.vi
PRL BUFFER Rearrange Timestamp Array.vi
PRL BUFFER Rearrange Complex 2D Array.vi
PRL BUFFER Rearrange Analog 2D Array.vi
PRL BUFFER Rearrange Digital 2D Array.vi
PRL DW Check Data CSG.vi
PRL DW Clean Data CSG 2D.vi
PRL DW Clean Data SGL 2D.vi
PRL UTIL Calculate Activity.vi
PRL UTIL Calculate Data Rate.vi
PRL UTIL Convert Command for Display.vi
PRL UTIL Get Real Timestamp.vi
PRL UTIL Split Command.vi
PRL STR Convert CSG Array to Strings.vi
PRL STR Convert SGL Array to String.vi
PRL TS Clip Array.vi
PRL TS Waiter.vi
PRL QUEUE Multiple Elements at Opposite End.vi
PRL QUEUE Multiple Elements.vi
PRL DW Check Data SGL.vi
PRL TS Get Real Timestamp.vi
PRL STR Convert Number to String.vi
PRL UTIL Get Control Value.vi
PRL SIGNAL Create Waveform.vi
PRL CALC Calculate Live Buffers in Access Buffer.vi
PRL Data Selector.vi
PRL Snapshooter.vi
Statnett PMU Recorder Light 0.3.vi
PRL Remote.vi



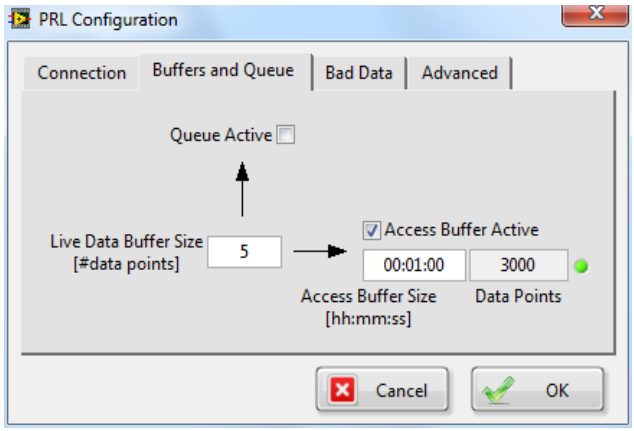
- PM omputer programs written in the Lab following
  - al-time PMU data
  - nections to PMUs and PDCs to be sent by the
  - a user can utilize to make a new program
  - al and internal documents.

PRL – Main GUI Handling Communications and Real-Time Data

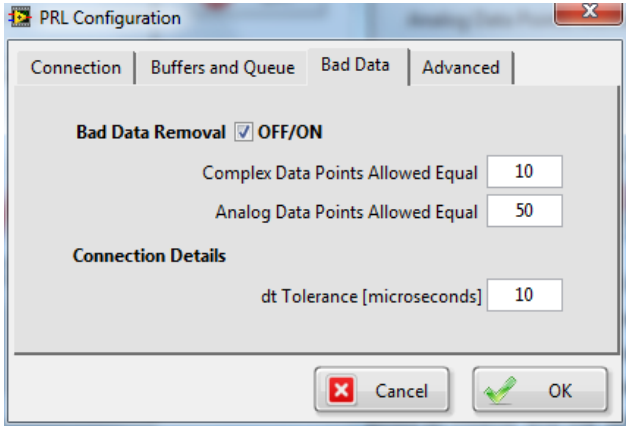
PRL Main GUI



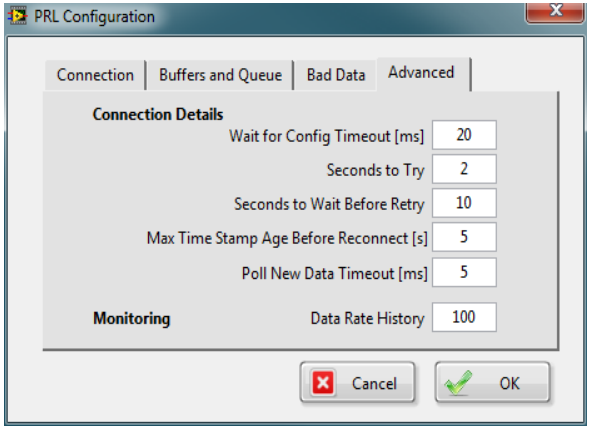
Connection Settings



Buffers and Queue

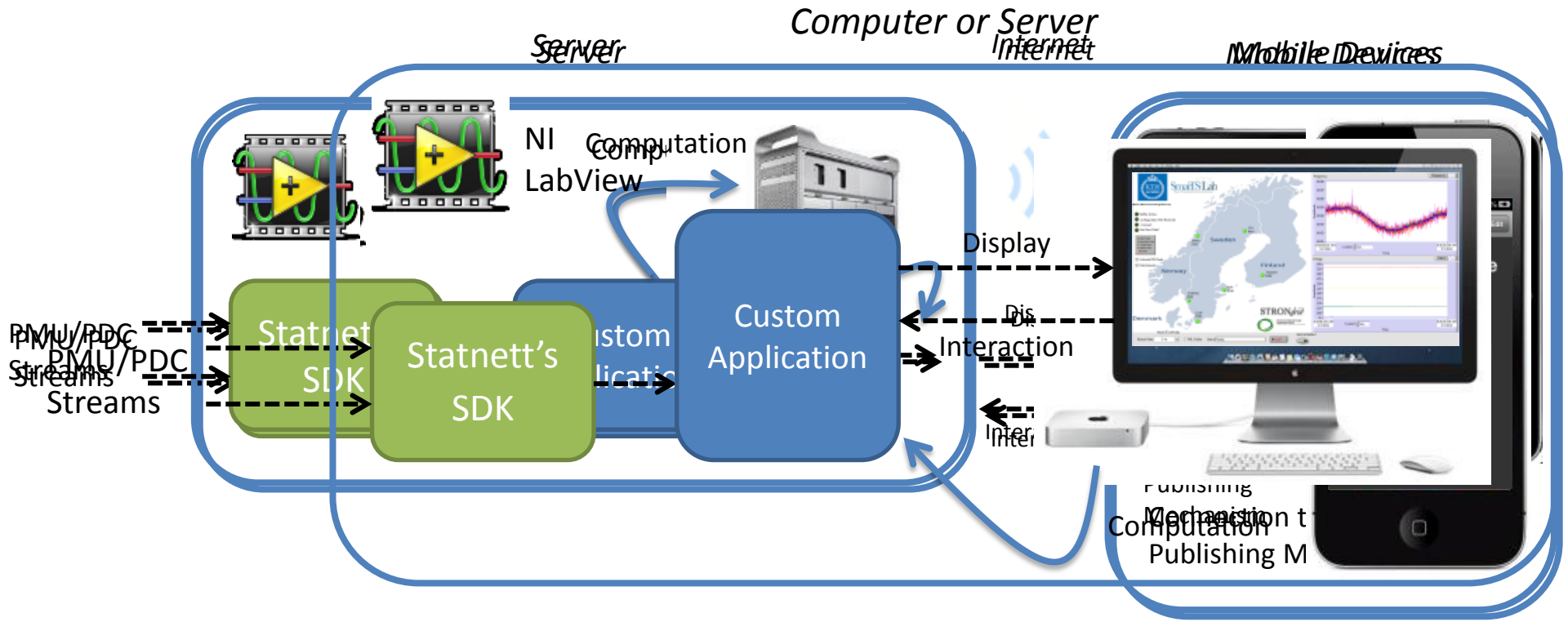


Bad Data



Advanced

SW Architecture Design supporting Modularity and Scalability for Different Deployment and Application Scenarios

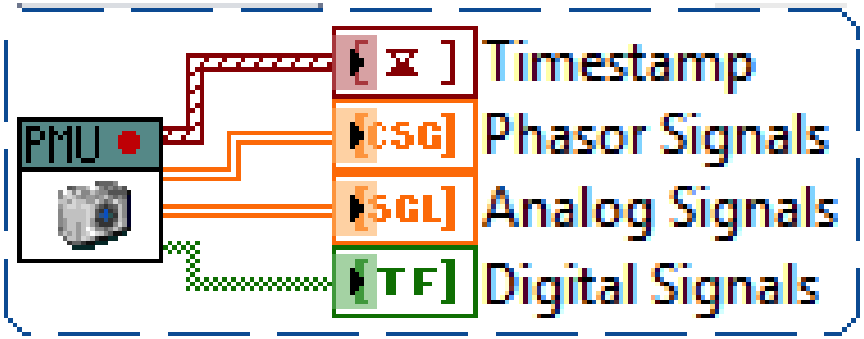


Tools

Use of the SDK for the App

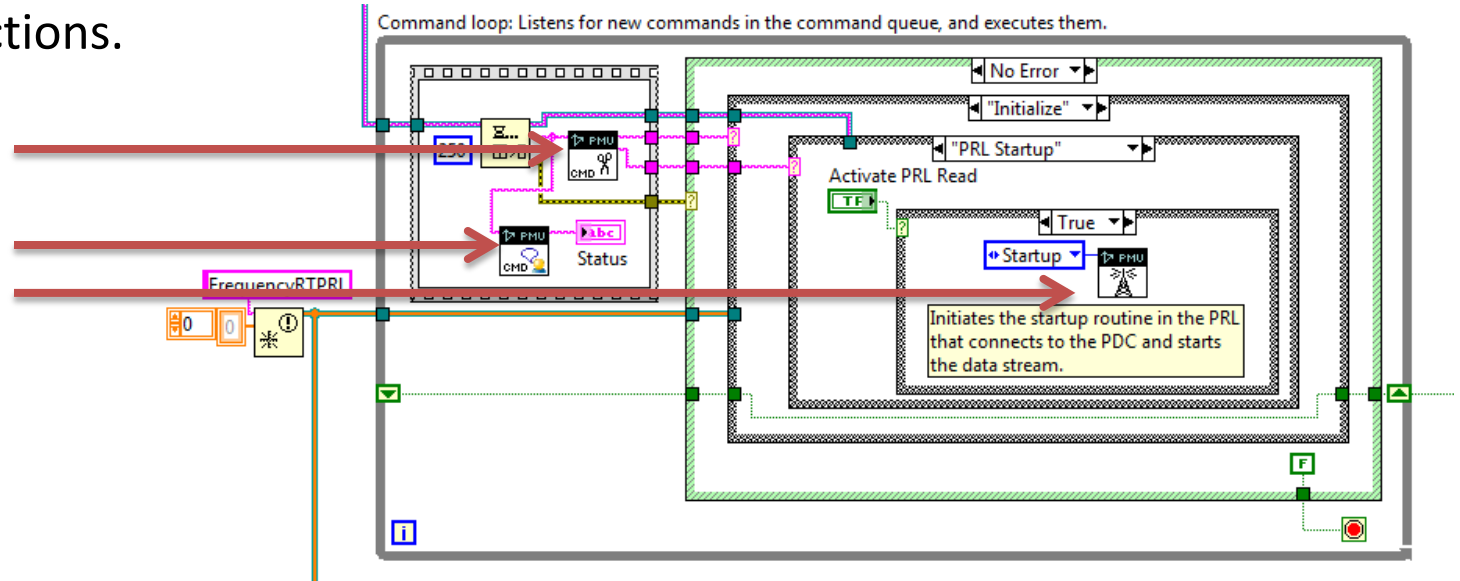
Tools
SW architecture and prototype implementation of the proposed method to provide transparently the required information / action

- Statnett's Synchrophasor SDK provides the measurements in LabView Datatypes



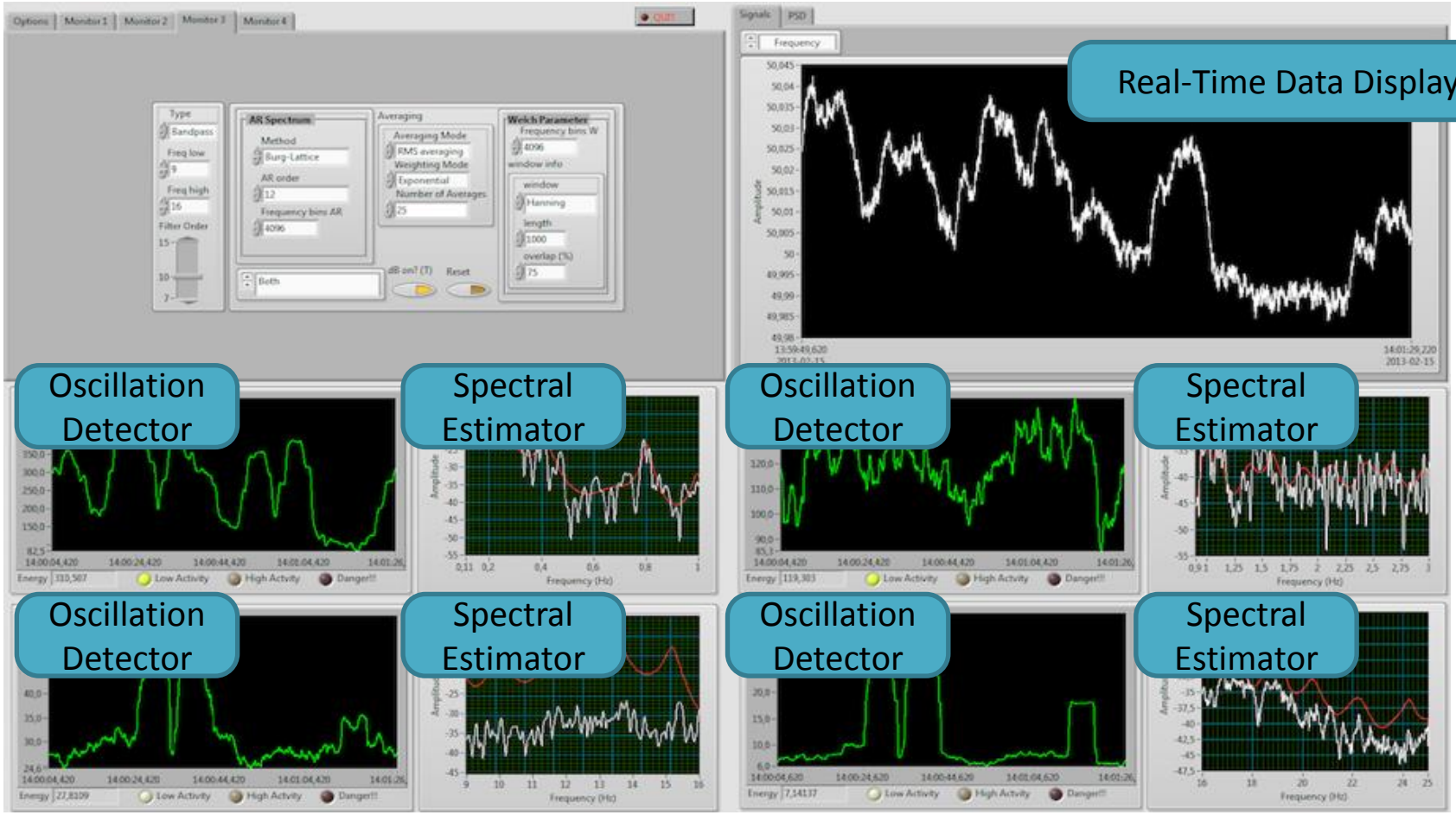
- The following figure shows a sub-set of the LabView implementation with PRL functions.

SDK Functions

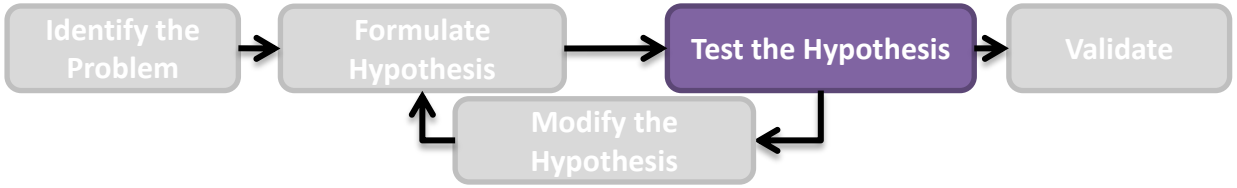

Command loop: Listens for new commands in the command queue, and executes them.
No Error
"Initialize"
"PRL Startup"
Activate PRL Read
True
Startup
Status
FrequencyRTPRL
Initiates the startup routine in the PRL that connects to the PDC and starts the data stream.

Tools

Implementation



Test the Hypothesis



Set-up Experimental Environment

Hardware and software needed to test the solution
Set-up the chain of real-time data acquisition

Design the Experiments Protocols

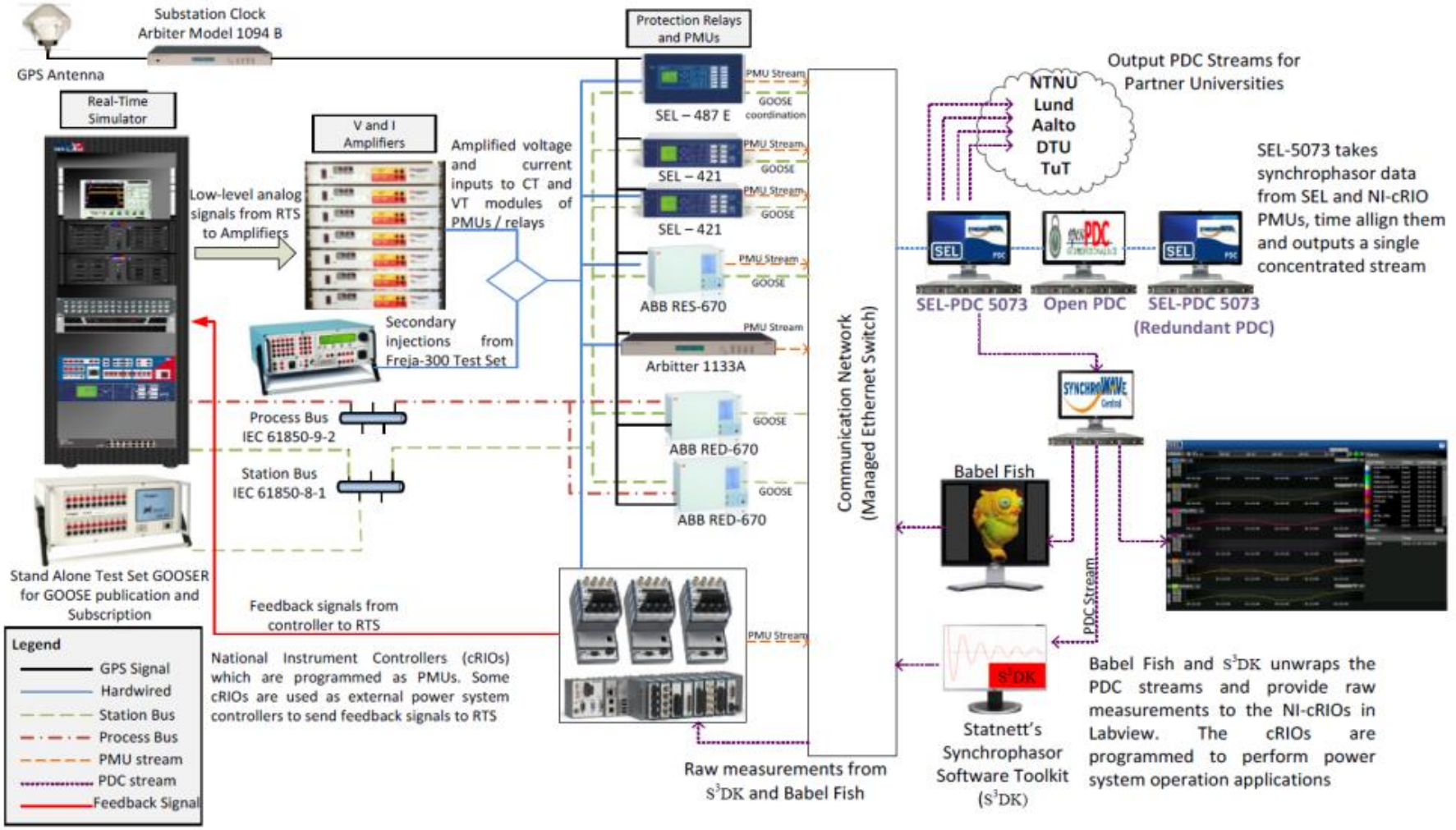
Develop an experiment protocol – i.e. what behaviour should be detected/control by the designed solution?

Models / HW Config.

Develop real-time simulation models that can reproduce the experiment protocol.
Set up hardware to reproduce the experiment protocol.

SmarTS Lab Architecture

Smart Transmission Systems Laboratory

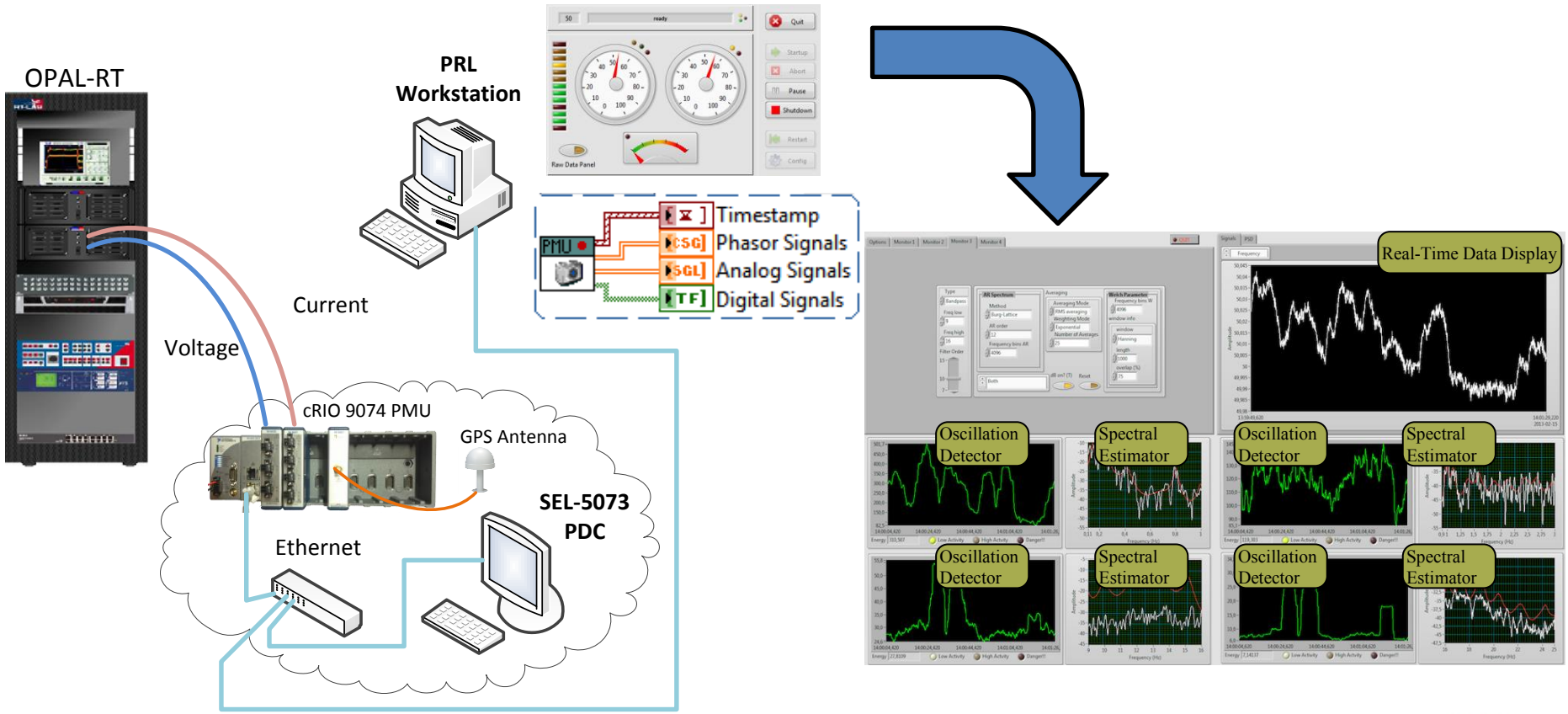

Substation Clock
Arbiter Model 1094 B
GPS Antenna
Real-Time Simulator
V and I Amplifiers
Low-level analog signals from RTS to Amplifiers
Amplified voltage and current inputs to CT and VT modules of PMUs / relays
Secondary injections from Freja-300 Test Set
Process Bus IEC 61850-9-2
Station Bus IEC 61850-8-1
Stand Alone Test Set GOOSER for GOOSE publication and Subscription
Feedback signals from controller to RTS
Protection Relays and PMUs
SEL – 487 E
SEL – 421
SEL – 421
ABB RES-670
Arbitter 1133A
ABB RED-670
ABB RED-670
PMU Stream
GOOSE
GOOSE coordination
Communication Network (Managed Ethernet Switch)
National Instrument Controllers (cRIOs) which are programmed as PMUs. Some cRIOs are used as external power system controllers to send feedback signals to RTS
Raw measurements from S³DK and Babel Fish
Output PDC Streams for Partner Universities
NTNU
Lund
Aalto
DTU
TuT
SEL-PDC 5073
Open PDC
SEL-PDC 5073 (Redundant PDC)
SEL-5073 takes synchrophasor data from SEL and NI-cRIO PMUs, time allign them and outputs a single concentrated stream
Babel Fish
PDC Stream
S³DK
Statnett's Synchrophasor Software Toolkit (S³DK)
Babel Fish and S³DK unwraps the PDC streams and provide raw measurements to the NI-cRIOs in Labview. The cRIOs are programmed to perform power system operation applications
Legend
GPS Signal
Hardwired
Station Bus
Process Bus
PMU stream
PDC stream
Feedback Signal

SmarTS Lab
Hardware Implementation

Set-up Experimental Environment

Hardware and software needed to test the solution

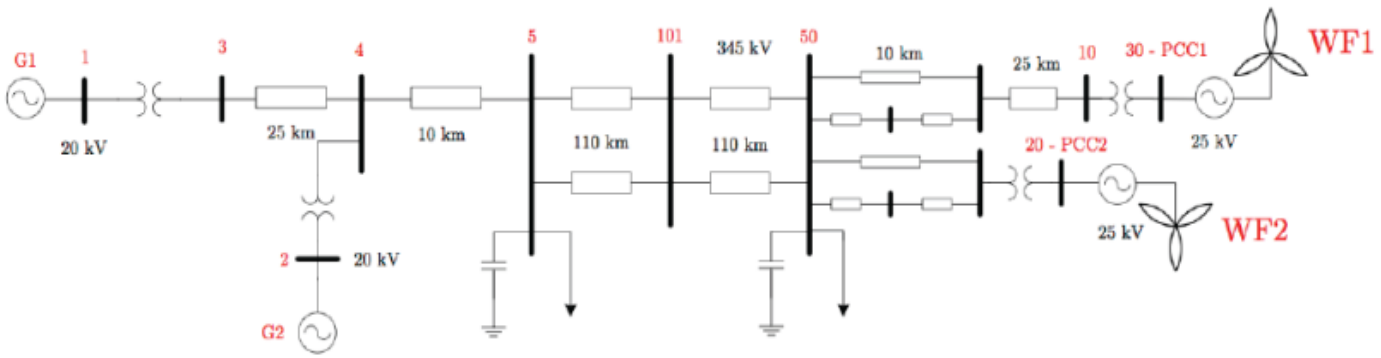
Integration of the RT HIL Simulation with the SDK and App



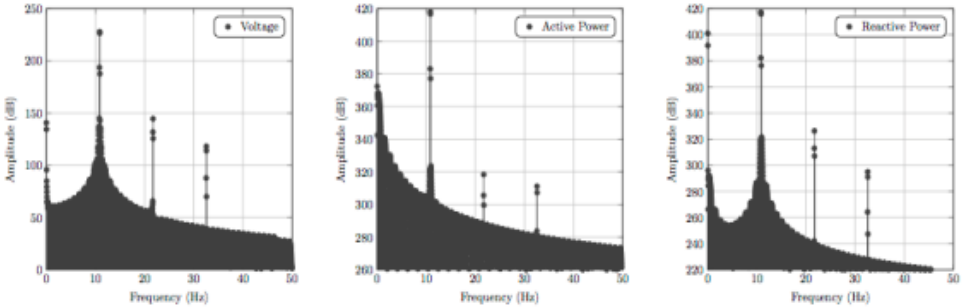
Models / HW Config.
Develop real-time simulation models that can reproduce the experiment protocol.

Real-Time Simulation Model

- Sensitivity study (not shown here) determined the main cause of the oscillations due to controller interactions (Grid side converter).
- Power System Model – Modified Klein-Rogers-Kundur system:
  - Loads vary with a Gaussian white noise input plus a configurable load portion



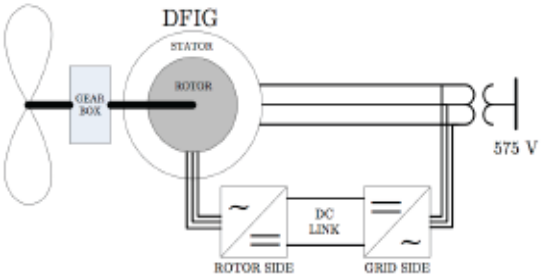
- Simulation spectrum – very similar to an actual power grid:

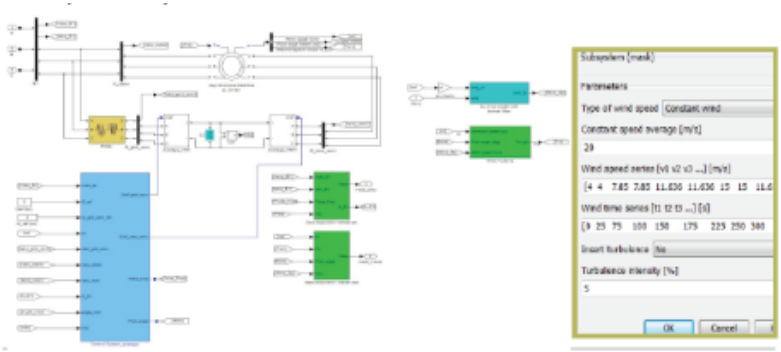


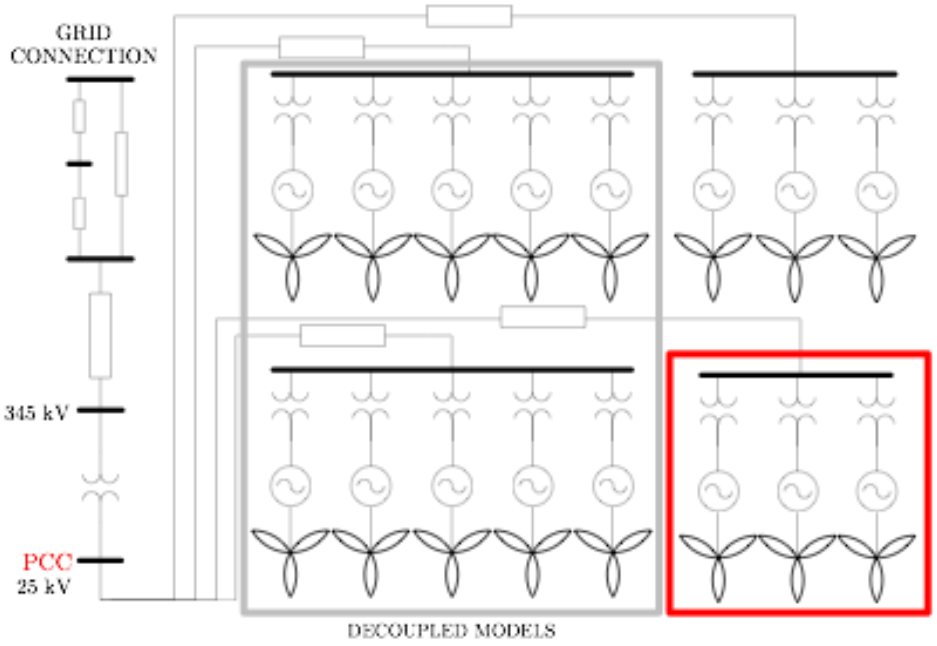
(a) Voltage (b) Active Power (c) Reactive Power

Wind Farm Models

- Farm 1 : 16 turbines Farm 2: 13 turbines
- DFIG (Type 3) aggregated turbines (4 to 6 turbines)

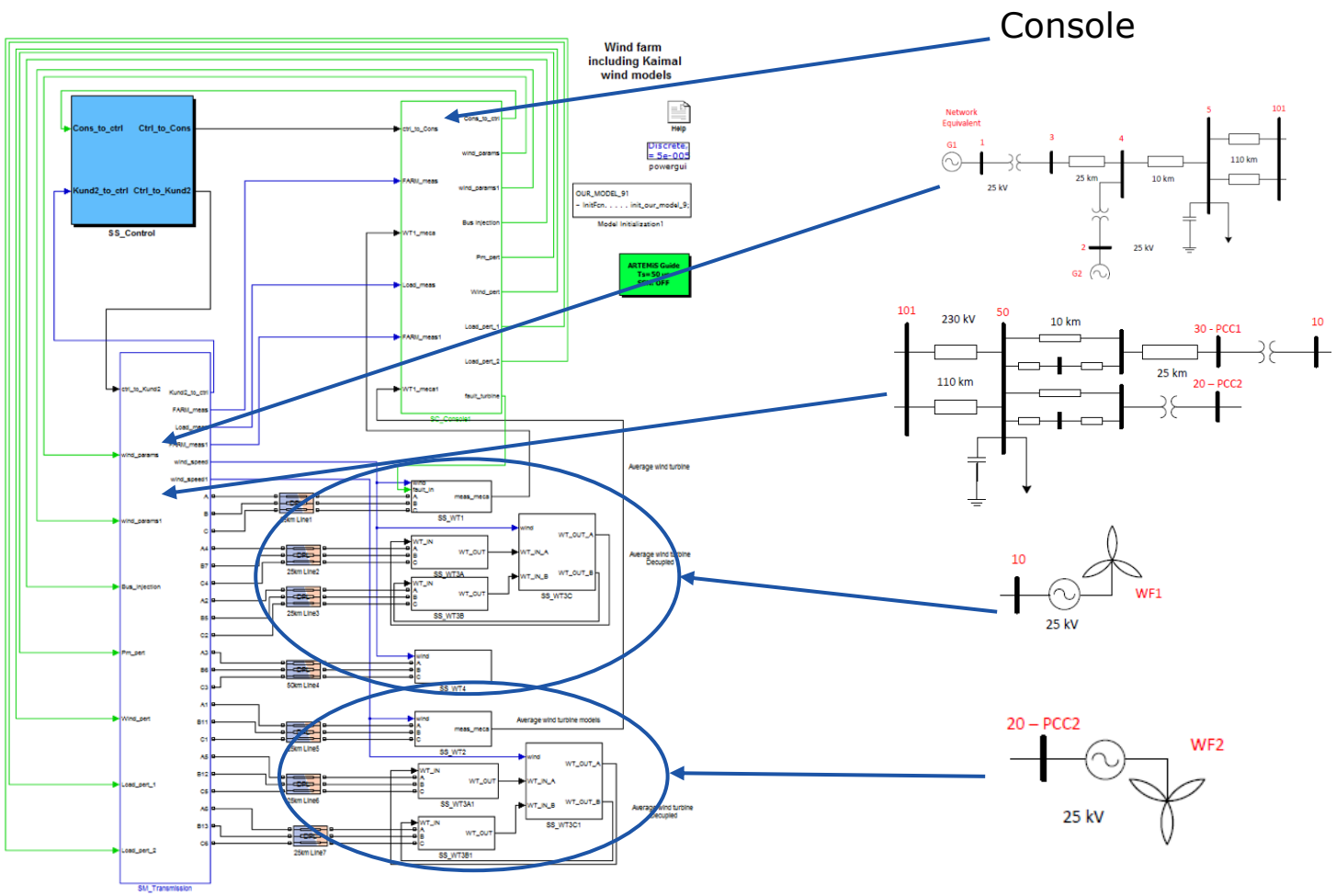






Power System Model for

Real-Time Excecution

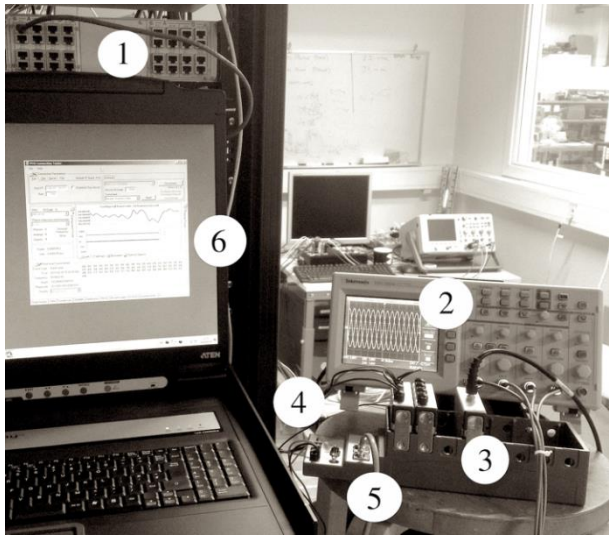


Set-up Experimental Environment
Set-up the chain of real-time data acquisition

RT Data Acquisition and Experiment Protocol

Design the Experiments Protocols
Develop an experiment protocol – i.e. what behaviour should be detected/control by the designed solution?

1. Opal RT Simulator
2. Oscilloscope connected to analog output
3. NI cRIO PMU
4. Voltage measurement module connected to analog output
5. Connection to network infrastructure for communication with PDC
6. PDC Server with Output Stream configured

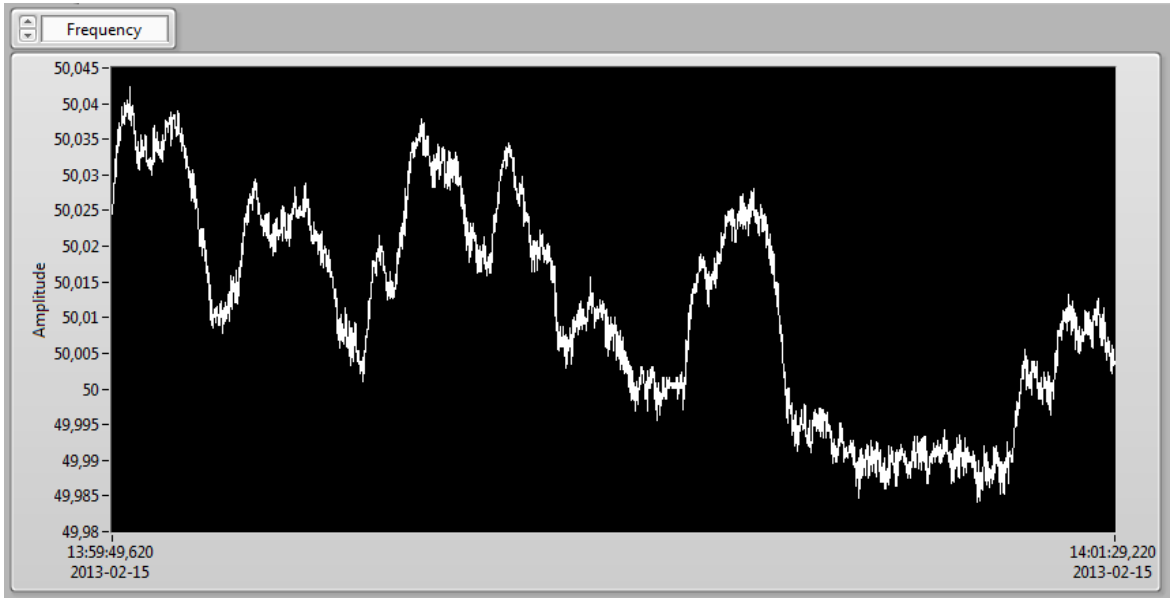


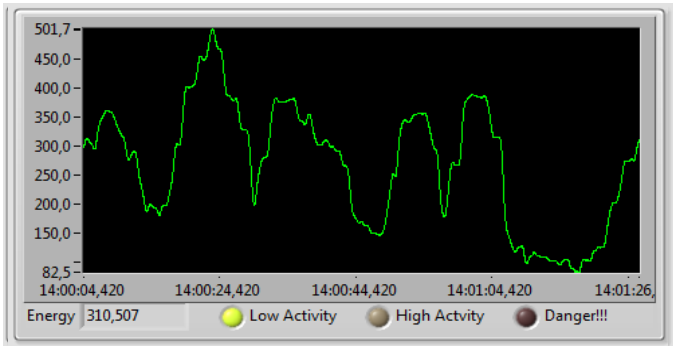
- Experiment Protocol Design:

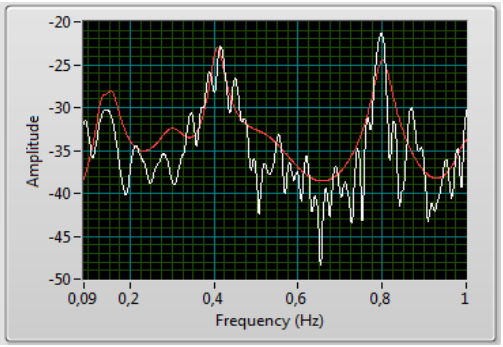
Start	Oscillation Injection	Additional Perturbations	Major Fault	End
Random load variation	Set perturbation at 10.83 Hz	Generating minor faults	Three phase Fault and line opening	End of the oscillation injection

Testing Results

Slow Dynamics

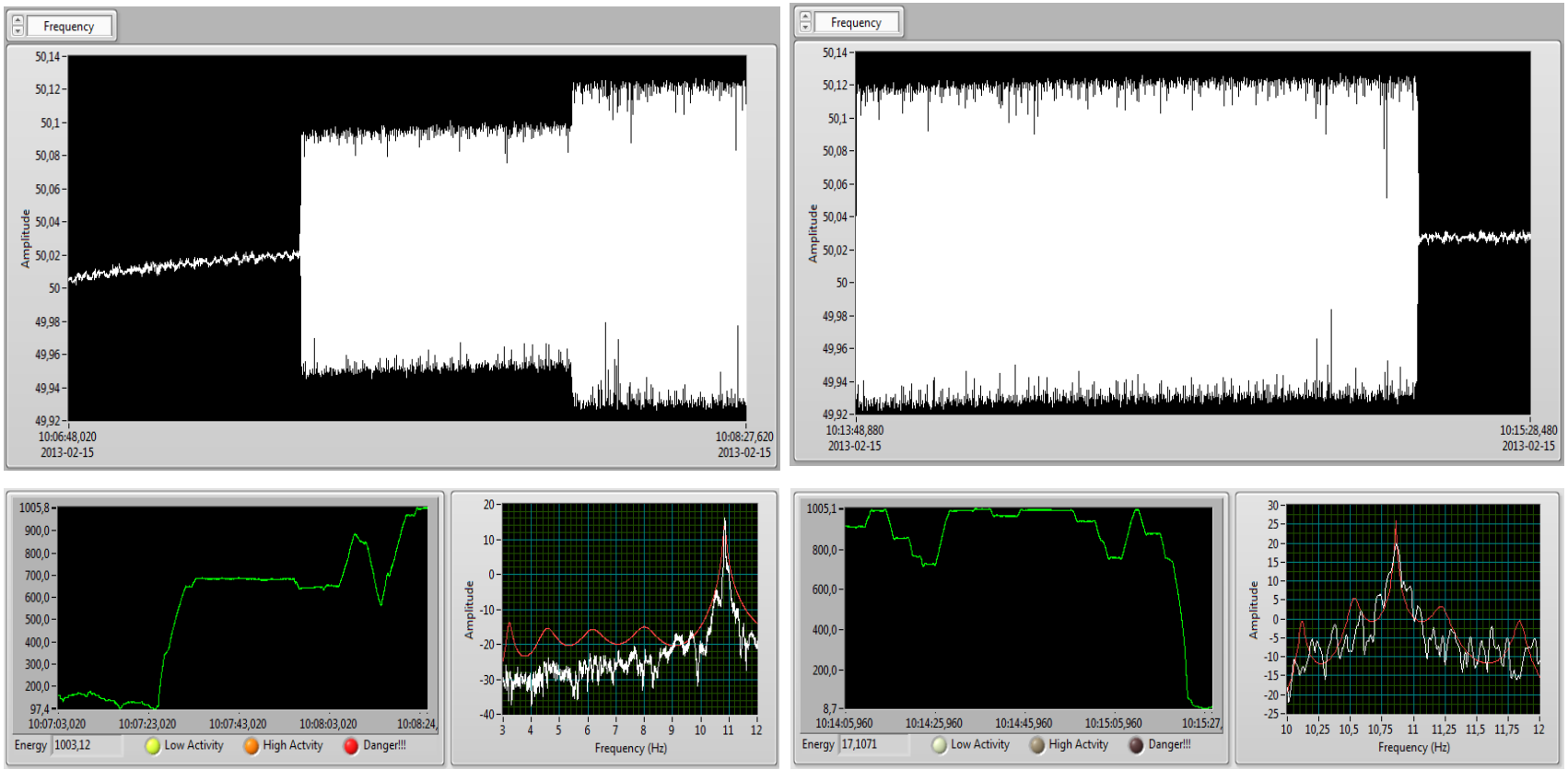




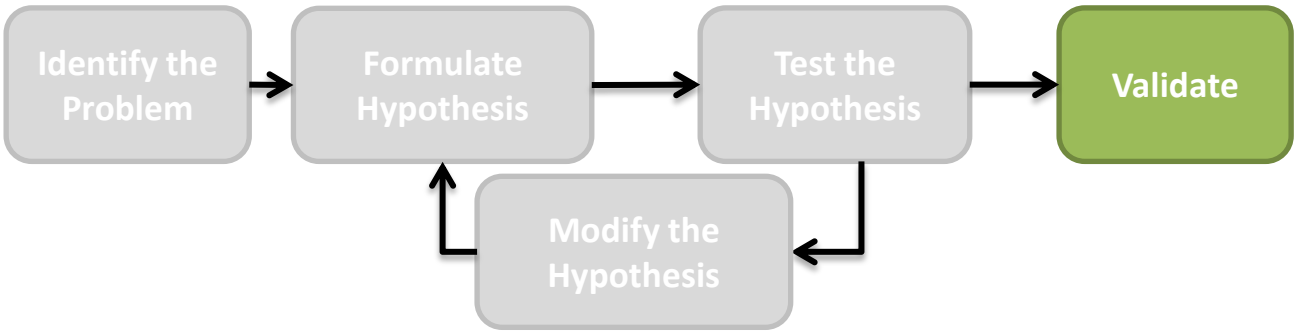


Testing Results

Oscillations at 10.83 Hz



Validate



Set-up Experimental Validation Environment

Configure the secondary experimental set-up for validation (different from the first) including the the chain of real-time data acquisition

Design the Validation Experiments Protocol

Develop new experiment protocols considering the properties of the validation environment

Gather Results and Determine Differences in Experiments

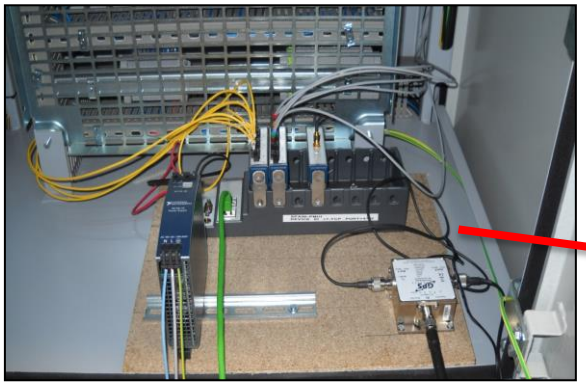
The PMU app should have performed well under the validation experiments. If not, find the source and go back to the hypothesis!

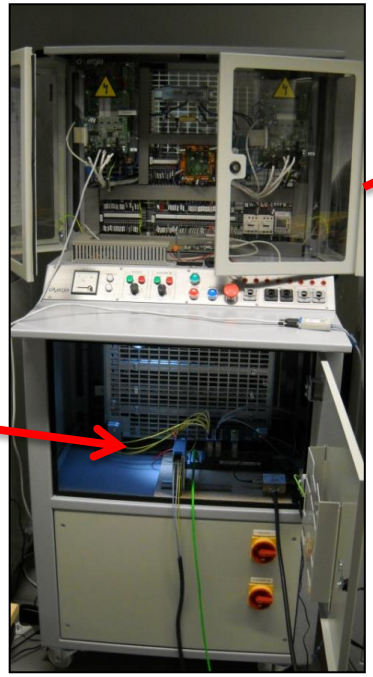
Set-up Experimental Validation Environment

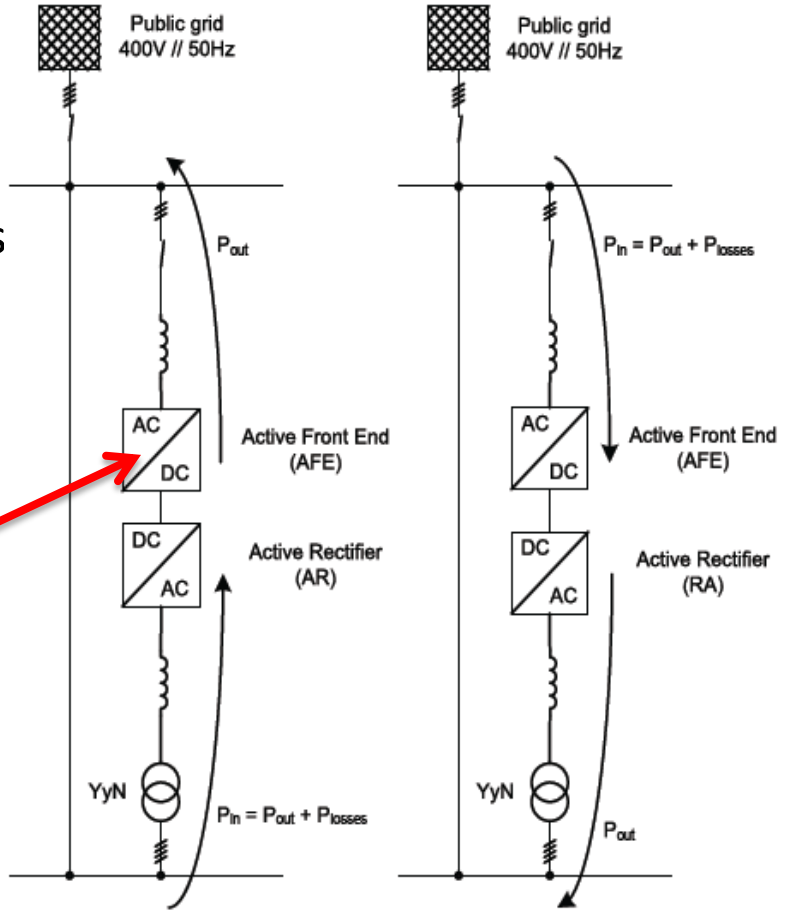
Configure the secondary experimental set-up for validation (different from the first) including the the chain of real-time data acquisition

Validation Set-Up

- Validation experiments were carried out on a grid emulator (Microgrid Lab.)
- Converters generate real power injections controlled by inverters

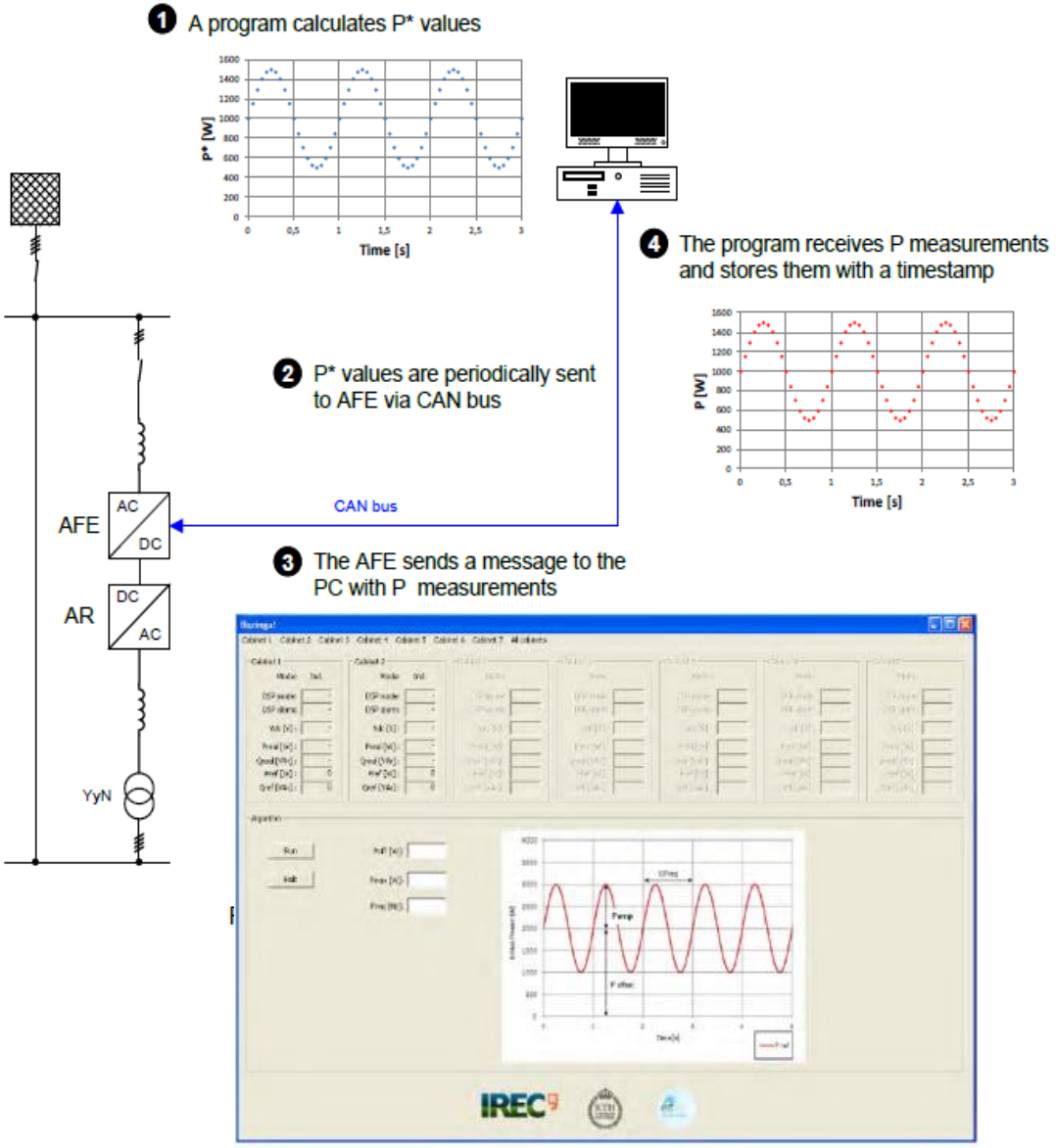


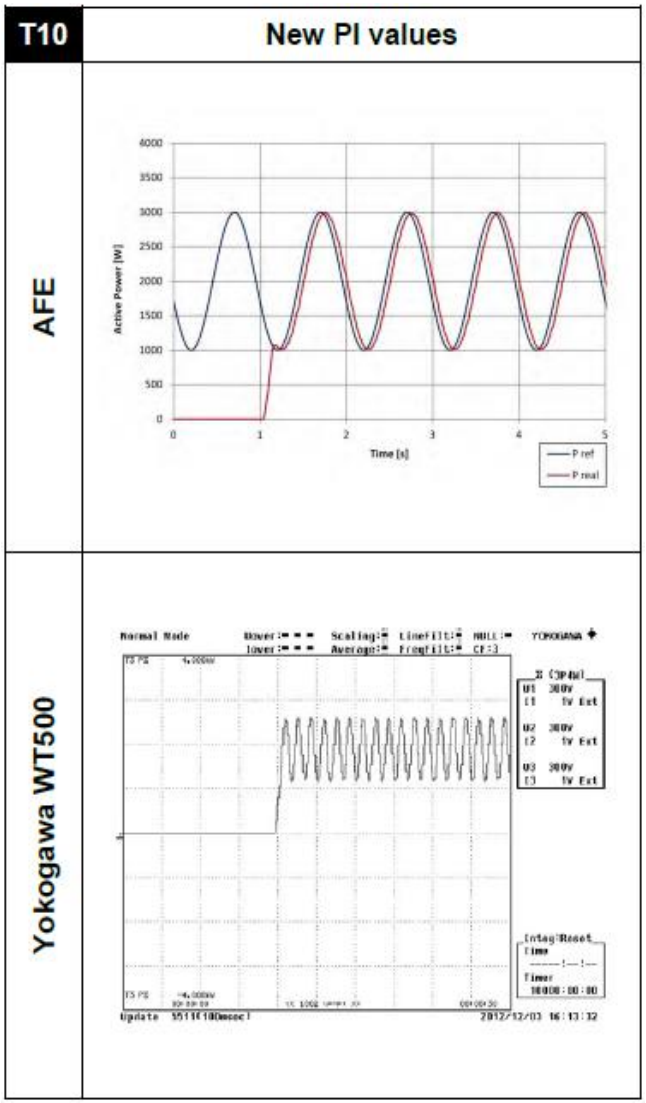




Validation Experiments Protocol

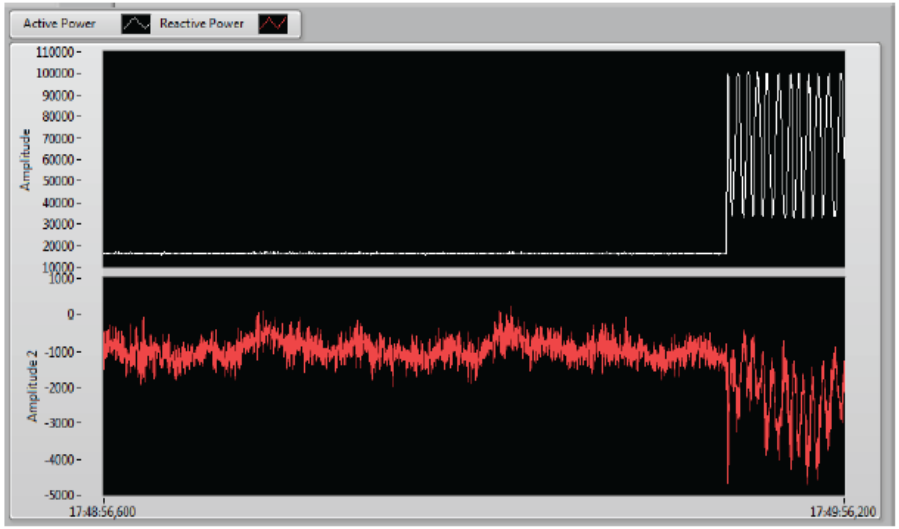
Validation Signal Generation and Injection



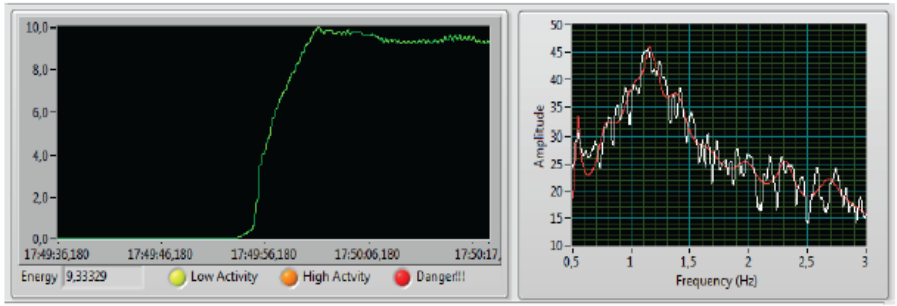


Validation Experiment Results

1Hz and 6 Hz oscillation injections

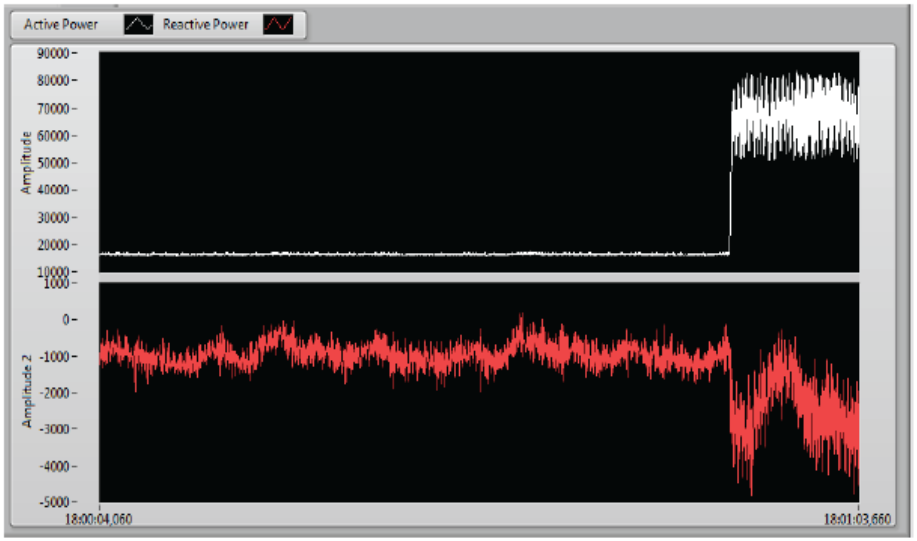


(a) Active and Reactive Powers

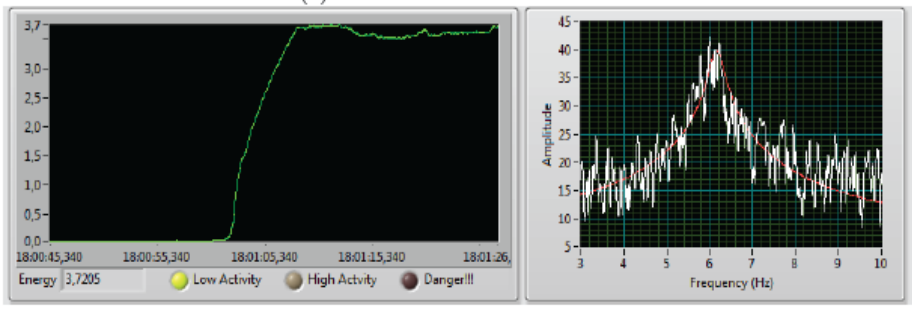


(b) Module

Figure 6.10: Screen shots of the *Monitoring Tool* at the beginning of the oscillations injection at 1 Hz



(a) Active and Reactive Powers



(b) Module

Figure 6.11: Screen shots of the *Monitoring Tool* at the beginning of the oscillations injection at 6 Hz

Lessons Learnt

- PMU App Development **in the lab.** requires a development method – the proposed method based on the scientific method is not a bad choice…
- Working in a RT HIL Lab and designing new applications:
  - It's a lot of fun… but also a lot of hard work.
  - Not necessarily will lead to journal papers… but you will learn more than you imagine (or be willing too…)
- It takes the work to a different level: stuff has to work!
  - The most important side-product: more experts!
- References:
  - On SmarTS Lab:
    - L. Vanfretti, et al, "SmarTS Lab: A laboratory for developing applications for WAMPAC systems," IEEE PES General Meeting 2012, San Diego, CA, USA.
  - On Statnett's Synchrophasor SDK:
    - L. Vanfretti, V. H. Aarstrand, M. Shoaib Almas, V. Peric and J. O. Gjerde, "A software development toolkit for real-time synchrophasor applications," IEEE PowerTech 2013, Grenoble, France.
  - On the Fast RT Tool:
    - L. Vanfretti, M. Baudette, J.L. Dominguez Garcia, I. Al-Khatib, M.S. Almas and J.O. Gjerde, "A PMU-Based Monitoring Application for Fast Real-Time Oscillation Detection from Wind Farm Interactions," to be submitted to Electric Power Systems Research.
  - On Testing Methods:
    - L. Vanfretti, M. Baudette, I. Al-Khatib, M. S. Almas, and J. O. Gjerde, "Testing and Validation of a Fast Real-Time Oscillation Detection PMU-Based Application for Wind-Farm Monitoring," Invited Paper, Technical Session, Track 5: Communication and Control in Smart Grids, in Proceedings of the First International Black Sea Conference on Communications and Networking 2013 (BlackSeaCom 2013), Batumi, Georgia.
  - On Validation Methods:
    - M. Baudette, L. Vanfretti, G. Del-Rosario, A. Ruiz-Alvarez, J.L. Dominguez Garcia, I. Al-Khatib, M. S. Almas, I. Cairo, and J.O. Gjerde, "Validating a Real-Time Application for Monitoring of Sub-Synchronous Wind Farm Oscillations," ISGT 2014, Washington DC.

Powered by S³DK
Statnett's SynchrophasorSDK

Thank you!

Questions?

luigiv@kth.se
luigi.vanfretti@statnett.no

KTH VETENSKAP OCH KONST

SmarTS Lab
Smart Transmission Systems Laboratory